

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การนำระบบ LAN มาประยุกต์กับระบบทะเบียนนักศึกษา

(LAN and Information System for Student Registration)



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของภาคการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาคณิตศาสตร์ประยุกต์

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2534

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LAN and Information System for Student Registration



Mr. Thongchai Lertmaneethaweesub
Mr. Paiboon Pattanasaksiri
Mr. Monchai Jarungruengkait
Mr. Sittichai Lertwattanak

A Special Project Submitted in Partial Fulfillment of the
Requirement for the Degree of Bachelor of Science
Department of Mathematics and Computer Sciences
Faculty of Science
King Mongkut's Institute of Technology Ladkrabang

ปัญหาพิเศษเรื่อง การนำระบบ LAN มาประยุกต์กับระบบทะเบียนนักศึกษา

LAN and Information System for Student Registration

ชื่อนักศึกษา

นายธงชัย เลิศมณีทวีทรัพย์

นายไพบุลย์ พัฒนศักดิ์ศิริ

นายมนต์ชัย จรุงเรืองเกียรติ

นายสิทธิชัย เลิศวัฒนารักษ์

ปัญหาพิเศษนี้ กรรมการสอบปัญหาพิเศษได้ตรวจพิจารณาแล้ว เห็นชอบแล้วจึงอนุมัติให้เป็นส่วนหนึ่ง ของการศึกษาตามหลักสูตร วิทยาศาสตร์บัณฑิต สาขาคณิตศาสตร์ประยุกต์และวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2534



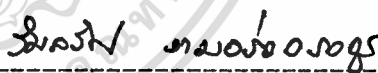
(รศ. วิเชียร ศรีเลื้อขาม)

หัวหน้าภาควิชา



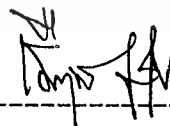
(ผู้ช่วยศาสตราจารย์ พิชรินทร์ เหมโชติ)

กรรมการการสอบปัญหาพิเศษ



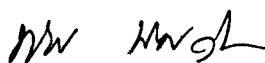
(อาจารย์ วิมลรัตน์ งามอร่ามวารังกูร)

กรรมการการสอบปัญหาพิเศษ



(นายเรืองฤทธิ์ เตือนทองสุข)

อาจารย์ที่ปรึกษาปัญหาพิเศษ



(อาจารย์ พิชรี เลิศวิจิตรศิลป์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ใช้ในเชิงพาณิชย์ด้านการค้า
อาจารย์ที่ปรึกษาปัญหาพิเศษ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงการพิเศษ การนำระบบ LAN มาประยุกต์กับระบบทะเบียนนักศึกษา

ชื่อนักศึกษา นายธงชัย เลิศมณีทวีทรัพย์
นายไพบุลย์ พัฒนศักดิ์ศิริ
นายมนต์ชัย จรุงเรือง เกียรติ
นายสิทธิชัย เลิศวัฒนารักษ์

อาจารย์ที่ปรึกษา นาย เรืองฤทธิ์ เตือนทองสุข
อาจารย์พี่ศรี เลิศวิจิตรศิลป์

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์

ปีการศึกษา 2534

บทคัดย่อ

ในปัจจุบัน ระบบ LAN ได้มีการนำมาใช้กันอย่างกว้างขวาง ไม่ว่าจะเป็นส่วนงานใด ๆ เพราะความที่ LAN สามารถใช้งานได้หลาย ๆ user พร้อม ๆ กัน หรือที่เราเรียกกันว่า Multiuser จึงทำให้ระบบงานต่าง ๆ มีความคล่องตัว รวดเร็วมากขึ้น ดังนั้นเราจึงได้นำระบบ LAN มาประยุกต์ใช้กับระบบงานทะเบียนนักศึกษา

การบริหารระบบงานทะเบียนนักศึกษา เป็นงานที่มีความซับซ้อน และมีขั้นตอนในการปฏิบัติการมาก การนำเอาระบบ LAN (Local Area Network) และคอมพิวเตอร์เข้ามาช่วยในการบริหารงานนั้น จะทำให้สามารถจัดเก็บข้อมูลได้อย่างมีประสิทธิภาพ และลดขั้นตอนในการปฏิบัติการได้มาก ในการทำปัญหาพิเศษเรื่องนี้จะมุ่งเน้นในการใช้งานให้ใช้งานได้ง่าย ลดขั้นตอนการทำงานและการนำระบบ LAN มาประยุกต์ใช้งานจะทำให้ข้อมูลถูกเก็บและจัดการอย่างเป็นระบบระเบียบ

เอกสารประกอบที่ส่งมอบไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไปว่ากรณิดจกั้สืบ ลึกขั้ห้ห้เจีให้ดัดแปลงแก้หา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Special Project Title LAN and Information System for Student
Registration

Name Mr. Thongchai Lertmaneethaweesub
 Mr. Paiboon Pattanasaksiri
 Mr. Monchai Jarungruengkait
 Mr. Sittichai Lertwattanak

Special Project Advisor Mr. Ruengrith Duenthongsuk
 Mrs. Patcharee Lertvichitsilp

Department Mathematics and Computer Sciences

Academic Year 1991

Abstract

At present, LAN (Local Area Network) is used widely in many jobs because the capability of LAN that can be used by many users simultaneously, is called Multiuser. LAN makes system better, thus we can apply LAN for the student registration.

The student registration is the complex system and has many jobs. To help in management LAN and computer can be used to store many data efficiently and can reduce process of jobs in registration. This special problem is specific for easy to use, decrease process and LAN will store and manage data correctly.

กิตติกรรมประกาศ

ปัญหาพิเศษเรื่อง การนำระบบ LAN มาประยุกต์กับระบบทะเบียนนักศึกษา นี้สำเร็จลงได้ด้วยความสะดวกเรียบร้อย ข้าพเจ้าในนาม คณะผู้จัดทำขอขอบคุณ บุคคลดังต่อไปนี้ ที่ได้ช่วยในเรื่องต่าง ๆ ให้สำเร็จลุล่วงโดยดี

1. นายเรืองฤทธิ์ เตือนทองสุข ผู้ริเริ่มโครงการพิเศษนี้และเป็นผู้ที่ให้การสนับสนุนช่วยเหลือมาโดยตลอด
2. อาจารย์พัชรี เลิศวิจิตรศิลป์ ผู้ที่ให้การช่วยเหลือในทุก ๆ ด้านและเป็นผู้ที่คอยว่ากล่าวตักเตือนให้พวกกระผมได้ดีและทำจนโครงการพิเศษนี้สำเร็จลงได้
3. ผู้ช่วยศาสตราจารย์ภคินี ยิมเรวัตติ ผู้ที่คอยดูแลให้ข้อคิดและคำแนะนำต่าง ๆ เกี่ยวกับโครงการพิเศษนี้ รวมทั้งหนังสือข้อมูลต่าง ๆ เกี่ยวกับทะเบียน
4. นายกุชงค์ เกษสุวรรณ ผู้ที่ให้คำแนะนำทางด้านการเขียนโปรแกรม
5. ห้อง ๆ ที่ภาควิชาที่ยอมมาช่วยบ่อนข้อมูลให้
6. เพื่อน ๆ ทุกคนที่คอยช่วยเหลือและให้กำลังใจ

สารบัญรูป

รูปที่	หน้า
2.1 แสดงลักษณะของสายส่งข้อมูลชนิดสายคู่บิดเกลียวแบบหุ้มฉนวน	2-4
2.2 แสดงลักษณะของสายส่งข้อมูลแบบโคแอกเชียล	2-5
2.3 แสดงลักษณะของสายส่งข้อมูลแบบเส้นใยนำแสง	2-6
2.4 แสดงรูปแบบกลุ่มข้อมูลสื่อสารของ IEEE 802.3 MAC	2-8
2.5 แสดงการใช้งานระบบ Ethernet โดยใช้สายโคแอกเชียลแบบบาง	2-9
2.6 แสดงการใช้งานระบบ Ethernet โดยใช้สายโคแอกเชียลแบบหนา	2-9
2.7 แสดงฮาร์ดแวร์ที่ใช้ในระบบเครือข่ายที่ใช้สายส่งข้อมูลแบบบาง	2-11
2.8 แสดงฮาร์ดแวร์ที่ใช้ในระบบ Ethernet ที่ใช้สายส่งข้อมูลแบบหนา	2-12
2.9 แสดงลักษณะของ BNC ต่างๆ	2-13
2.10 แสดงการใช้งานระบบเครือข่าย ARCnet	2-15
2.11 แสดงฮาร์ดแวร์ของระบบเครือข่ายแบบ ARCnet	2-16
2.12 แสดงการใช้งานระบบเครือข่าย IBM Token-ring	2-17
2.13 แสดงฮาร์ดแวร์ที่ใช้ในระบบเครือข่าย IBM Token-ring	2-18
2.14 แสดงการทำงานของระบบเครือข่ายแบบ Token-Passing Bus	2-19
2.15 แสดงการใช้งานระบบเครือข่าย starLAN	2-20
2.16 แสดงการใช้งานระบบเครือข่ายแบบ starLAN โดยใช้ daisy chain	2-20
2.17 แสดงปรัชญาการออกแบบ Netware V3.11	2-23
2.18 แสดงโปรโตคอลของ Netware V3.11	2-23
2.19 แสดงลักษณะของการเข้าถึงข้อมูลในแบบ disk mirroring	2-26
2.20 แสดงลักษณะของการเข้าถึงข้อมูลในแบบ disk duplexing	2-26
2.21 แสดงการเชื่อมโยงระบบเครือข่ายในลักษณะ external router	2-28
2.22 แสดงการเชื่อมโยงระบบเครือข่ายในลักษณะ internal router	2-29
2.23 แสดง shell ของ Netware ทำหน้าที่ติดต่อกับระหว่าง Dos ของ สถานี	

2.24 แสดงการทำงานของ Dos (สภาวะแวดล้อมแบบมีผู้ใช้คนเดียว)	2-35
2.25 แสดงการทำงานของ Netware (สภาวะแวดล้อมแบบมีผู้ใช้หลายคน)	2-36
2.26 แสดงการทำงานของ shell ภายใต้ Netware	2-37
2.27 แสดงการสนับสนุนการใช้ Dos ภายใต้ Netware	2-37
2.28 แสดงรูปโทโพโลยีแบบดาว	2-39
2.29 แสดงรูปโทโพโลยีแบบบัส	2-40
2.30 แสดงรูปโทโพโลยีแบบวงแหวน	2-41
2.31 แสดงรูปโทโพโลยีแบบวงแหวน-ดาว	2-42
2.32 แสดงตัวอย่างของ OSI Model	2-47
2.33 แสดง command/response protocol on bus and ring	2-48
2.34 แสดงรูป interrupt driven protocol	2-50
2.35 แสดงรูป token-passing protocol	2-52
2.36 แสดงกรณีที่เกิด collision ของ CSMA/CD protocol	2-55
3.1 แสดงหน้าจอเมนูหลักของฝ่ายทะเบียน	3-25
3.2 แสดงหน้าจอเมนูของประวัตินักศึกษา	3-26
3.3 แสดงหน้าจอเมนูของการบันทึกรายวิชา	3-27
3.4 แสดงหน้าจอเมนูของการลงทะเบียนนักศึกษา	3-28
3.5 แสดงหน้าจอเมนูผลการเรียนของนักศึกษา	3-29
3.6 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา	3-30
3.7 แสดงหน้าจอประวัติการศึกษาของนักศึกษา	3-31
3.8 แสดงหน้าจอของรายละเอียดวิชาบังคับ	3-32
3.9 แสดงหน้าจอของรายละเอียดวิชาเลือกเฉพาะภาคการศึกษา	3-33
3.10 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนในเทอมนี้	3-34
3.11 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนทั้งหมดในหลักสูตร	3-35
3.12 แสดงหน้าจอของโครงสร้างหลักสูตรทั้งหมด	3-36
3.13 แสดงหน้าจอของการกำหนดช่วงการลงทะเบียน	3-37
3.14 แสดงหน้าจอการลงทะเบียนนักศึกษา	3-38

3.15	แสดงหน้าจอเมื่อต้องการบอณาผลการศึกษา	3-39
3.16	แสดงหน้าจอของการแก้ไขเฉพาะรายวิชา	3-40
3.17	แสดงหน้าจอการสรุปผลการศึกษาเฉพาะภาคเรียน	3-41
3.18	แสดงรายการงานของส่วนอาจารย์	3-42
3.19	แสดงงานในส่วนของเกรดนักศึกษา	3-43
3.20	แสดงการทำงานในส่วนของการเปลี่ยนแปลงแก้ไขผลการศึกษาเป็นรายวิชา	3-44
3.21	แสดงหน้าจอของการเปลี่ยนแปลงแก้ไขผลการศึกษาของนักศึกษา	3-45
3.22	แสดงหน้าจอของการตรวจสอบข้อมูลของนักศึกษา	3-46
3.23	แสดงหน้าจอของประวัติส่วนตัวนักศึกษา	3-47
3.24	แสดงหน้าจอประวัติการศึกษาของนักศึกษา	3-48
3.25	แสดงหน้าตรวจสอบการลงทะเบียนนักศึกษา	3-49
3.26	แสดงรายการงานของส่วนนักศึกษา	3-50
3.27	แสดงหน้าจอให้บอณารหัสนักศึกษาเพื่อให้แสดงประวัติของนักศึกษา	3-51
3.28	แสดงประวัติของนักศึกษา	3-52
3.29	แสดงรายการย่อยของการตรวจสอบผลการศึกษา	3-53
3.30	แสดงการตรวจสอบผลการศึกษาเป็นรายวิชา	3-54
3.31	แสดงหน้าจอการตรวจสอบผลการศึกษาเป็นภาคการเรียน	3-55
3.32	แสดงการลงทะเบียนเรียน	3-56
3.33	แสดงรายการย่อยของการตรวจสอบรายละเอียดวิชาเรียน	3-57
3.34	แสดงรายละเอียดวิชาเรียนที่เป็นวิชาบังคับของภาคการเรียน	3-58
3.35	แสดงรายละเอียดวิชาเรียนที่เป็นวิชาเลือกของภาคการเรียน	3-59
3.36	แสดงรายละเอียดวิชาที่เปิดสอนในปัจจุบัน	3-60
3.37	แสดงรายละเอียดวิชาเรียนทั้งหมดในหลักสูตร	3-61
4.1	แสดงหน้าจอเมนูหลักของฝ่ายทะเบียน	4-9
4.2	แสดงหน้าจอเมนูของประวัตินักศึกษา	4-10
4.3	แสดงหน้าจอเมนูของการบันทึกรายวิชา	4-11

4.4 แสดงหน้าจอเมนูของการลงทะเบียนนักศึกษา	4-12
4.5 แสดงหน้าจอเมนูผลการเรียนของนักศึกษา	4-13
4.6 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา	4-14
4.7 แสดงหน้าจอประวัติการศึกษาของนักศึกษา	4-15
4.8 แสดงหน้าจอของรายละเอียดวิชาบังคับ	4-16
4.9 แสดงหน้าจอของรายละเอียดวิชาเลือกเฉพาะภาคการศึกษา	4-17
4.10 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนในเทอมนั้น	4-18
4.11 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนทั้งหมดในหลักสูตร	4-19
4.12 แสดงหน้าจอของโครงสร้างหลักสูตรทั้งหมด	4-20
4.13 แสดงหน้าจอของการกำหนดช่วงการลงทะเบียน	4-21
4.14 แสดงหน้าจอการลงทะเบียนนักศึกษา	4-22
4.15 แสดงหน้าจอเมื่อต้องการบ่อนผลการเรียน	4-23
4.16 แสดงหน้าจอของการแก้ไขเฉพาะรายวิชา	4-24
4.17 แสดงหน้าจอการสรุปผลการเรียนเฉพาะภาคเรียน	4-25
4.18 แสดงรายการงานของส่วนอาจารย์	4-28
4.19 แสดงงานในส่วนของเกรดนักศึกษา	4-29
4.20 แสดงการทำงานในส่วนของการเปลี่ยนแปลงแก้ไขผลการเรียนเป็นรายวิชา	4-30
4.21 แสดงหน้าจอของการเปลี่ยนแปลงแก้ไขผลการเรียนของนักศึกษา	4-31
4.22 แสดงหน้าจอของการตรวจสอบข้อมูลของนักศึกษา	4-32
4.23 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา	4-33
4.24 แสดงหน้าจอประวัติการศึกษาของนักศึกษา	4-34
4.25 แสดงหน้าตรวจสอบการลงทะเบียนนักศึกษา	4-35
4.26 แสดงรายการงานของส่วนนักศึกษา	4-39
4.27 แสดงหน้าจอให้บ่อนรหัสนักศึกษาเพื่อให้เห็นประวัติของนักศึกษา	4-40
4.28 แสดงประวัติของนักศึกษา	4-41
4.29 แสดงรายการย่อยของการตรวจสอบผลการเรียน	4-42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

4.30	แสดงการตรวจสอบผลการศึกษาเป็นรายวิชา	4-43
4.31	แสดงหน้าจอกการตรวจสอบผลการศึกษาเป็นภาคการเรียน	4-44
4.32	แสดงการลงทะเบียนเรียน	4-45
4.33	แสดงรายการย่อยของการตรวจสอบรายละเอียดวิชาเรียน	4-46
4.34	แสดงรายละเอียดวิชาเรียนที่เป็นวิชาบังคับของภาคการเรียน	4-47
4.35	แสดงรายละเอียดวิชาเรียนที่เป็นวิชาเลือกของภาคการเรียน	4-48
4.36	แสดงรายละเอียดวิชาที่เปิดสอนในปัจจุบัน	4-49
4.37	แสดงรายละเอียดวิชาเรียนทั้งหมดในหลักสูตร	4-50



สารบัญ

บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญรูป	ฅ
บทที่ 1 บทนำ	1-1
1.1 ความสำคัญและที่มาของปัญหา	1-1
1.2 วัตถุประสงค์ของปัญหาพิเศษ	1-2
1.3 ขอบเขตของปัญหาพิเศษ	1-2
1.4 วิธีการดำเนินการ	1-2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	1-3
บทที่ 2 ระบบ LAN	2-1
2.1 บทนำ	2-1
2.2 องค์ประกอบของ LAN	2-2
2.2.1 อุปกรณ์ด้านฮาร์ดแวร์	2-2
- Microcomputer ในระบบ LAN	2-3
- ตัวกลางนำข้อมูล	2-4
- Network Interface Card	2-7
- Ethernet	2-7
- ARCnet	2-13
- token-ring	2-17
- token-bus	2-18
- starLAN	2-19
2.2.2 โปรแกรมควบคุมระบบ	2-21
- ระบบปฏิบัติการ Netware	2-21
- ความเป็นมาของ Netware	2-22

บทที่ 1 บทนำ

1.1 ความสำคัญและที่มาของปัญหาพิเศษ

ระบบธุรกิจในปัจจุบันนี้ ข้อมูลต่างๆ ถือว่ามีความจำเป็นในการตัดสินใจเกี่ยวกับการดำเนินงานทางธุรกิจ ในกรณีที่ระบบนั้นประกอบด้วยองค์กรหลายๆ องค์กร การสื่อสารข้อมูลถึงกันจึงมีความจำเป็นอย่างมาก ดังนั้นการนำเอาระบบคอมพิวเตอร์และระบบสื่อสารข้อมูลที่ทันสมัยเข้ามาใช้งาน ย่อมจะมีผลต่อระบบธุรกิจ ทำให้ระบบนั้นๆ ดำเนินงานได้ทันต่อเหตุการณ์และได้เปรียบคู่แข่งขันในเชิงการค้า และในปัจจุบันไม่เฉพาะในระบบธุรกิจเท่านั้น ในระบบงานอื่นๆ ที่เกี่ยวข้องกับ การใช้ข้อมูล ก็มีความจำเป็นเช่นเดียวกัน โดยที่จะเน้นในแง่ของ การเพิ่มความรวดเร็วในการทำงาน การใช้ทรัพยากรคอมพิวเตอร์อย่างคุ้มค่าและประหยัด การเชื่อมโยงการสื่อสารข้อมูลระหว่างผู้ใช้ให้ทำงานร่วมกันได้ในลักษณะที่สามารถติดต่อและส่งข้อมูลถึงกันและกันได้ การทำงานในลักษณะนี้เป็นการทำงานในระบบที่เราเรียกว่า "ระบบเครือข่าย (Network)" ซึ่งในปัจจุบันมีระบบเครือข่ายดังนั้นอยู่หลายแบบด้วยกัน แต่ที่นับว่าเป็นที่นิยมกันได้แก่ "ระบบเครือข่ายท้องถิ่น (Local Area Network)" หรือเรียกกันสั้นๆ ว่า "LAN"

LAN ถือเป็นระบบเครือข่ายที่มีบทบาทอย่างมากในวงการไมโครคอมพิวเตอร์ เนื่องจากว่าไมโครคอมพิวเตอร์ เป็นระบบคอมพิวเตอร์ที่ถูกออกแบบเพื่อใช้งานสำหรับผู้ใช้เพียงคนเดียว แต่ "LAN" เป็นระบบการเชื่อมโยงไมโครคอมพิวเตอร์หลายๆ เครื่องเข้าด้วยกันโดยที่มีคอมพิวเตอร์เครื่องหนึ่งทำหน้าที่จัดการสื่อสารข้อมูลต่างๆ ของคอมพิวเตอร์ทั้งหมดที่มาเชื่อมโยงกัน ข้อเด่นชัดที่แตกต่างจากการใช้งานคอมพิวเตอร์ที่ไม่มีการเชื่อมโยง ในลักษณะที่ ผู้ใช้หลายคน ในระบบ "LAN" สามารถใช้ข้อมูลร่วมกันได้ อีกทั้งมีการกำหนดสิทธิในการใช้งานข้อมูลอีกด้วย

ในระบบทะเบียนนักศึกษา เมื่อเริ่มมีการนำไมโครคอมพิวเตอร์มาใช้ จะเป็นในลักษณะที่ปราศจากการเชื่อมต่อคอมพิวเตอร์ร่วมกัน ดังนั้น การเปลี่ยนแปลงแก้ไข การดำเนินการอื่นๆ ที่เกี่ยวกับข้อมูลนักศึกษา ย่อมทำให้ระบบทะเบียนเกิดความล่าช้าได้

การนำ LAN เข้าประยุกต์ใช้งานในระบบทะเบียนนักศึกษาก็จะเป็นการทดสอบว่าสามารถทำให้ระบบทะเบียนมีความเร็วในการทำงานเพิ่มขึ้น และสามารถใช้งานข้อมูลนักศึกษาร่วมกันได้ ระหว่างเจ้าหน้าที่ทะเบียน อาจารย์ และนักศึกษาหรือไม่ โดยจะเป็นในลักษณะที่ดำเนินการศึกษาเรื่องราวต่างๆ เกี่ยวกับ LAN และเขียนโปรแกรมคอมพิวเตอร์บนระบบเครือข่าย LAN นั้นเอง

1.2 วัตถุประสงค์ของปัญหาพิเศษ

ศึกษาการทำงานต่างๆ ภายในระบบ LAN โดยเน้นกรณีการใช้ข้อมูลร่วมกัน อีกทั้งทดสอบว่าโปรแกรมที่เขียนขึ้นสามารถใช้งานในระบบ LAN ได้ และตรงกับ concept ของ LAN

1.3 ขอบเขตของปัญหาพิเศษ

1. ศึกษาการใช้ข้อมูลร่วมกัน และการสื่อสารข้อมูลภายในระบบ LAN
2. ศึกษาระบบทะเบียนนักศึกษาว่าประกอบด้วยส่วนของ นักศึกษา เจ้าหน้าที่ทะเบียน อาจารย์ เป็นในลักษณะ การลงทะเบียนเรียน ทะเบียนประวัติ ผลการศึกษา
3. ทดสอบโปรแกรมที่ทำขึ้นกับระบบ LAN เพื่อศึกษาความสามารถช่วยในระบบทะเบียนนักศึกษาได้มากน้อยเพียงใด

1.4 วิธีการดำเนินการ

1. ศึกษารายละเอียดต่างเกี่ยวกับระบบ LAN
2. ศึกษาความสามารถในการสื่อสารข้อมูลในระบบ LAN
3. ศึกษาความสามารถในการใช้ข้อมูลร่วมกันในระบบ LAN
4. วิเคราะห์ปัญหาและการออกแบบระบบทะเบียนนักศึกษา
5. ศึกษาความเป็นไปได้และข้อได้เปรียบในการนำระบบ LAN เข้ามาประยุกต์ระบบทะเบียนนักศึกษา
6. ศึกษาวิธีการเขียนโปรแกรม (ภาษา C และ BTRIEVE)
7. สร้างโปรแกรมเพื่อใช้กับระบบทะเบียนนักศึกษา โดยสามารถทำงานบนระบบ LAN ได้
8. จัดทำคู่มือประกอบสำหรับการใช้โปรแกรม
9. ทำการทดลองใช้โปรแกรม รวมทั้ง เปรียบเทียบกับระบบเดิม

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เข้าใจถึงความเป็นมาและรายละเอียดต่างๆ เกี่ยวกับระบบเครือข่ายท้องถิ่น (LAN)
2. สามารถนำเอาระบบ LAN มาประยุกต์ใช้งานได้
3. เข้าใจถึงหลักการและปัญหาที่เกิดขึ้นกับระบบทะเบียนนักศึกษา
4. โปรแกรมที่เขียนขึ้นสามารถทำงานบนระบบ LAN ได้
5. โปรแกรมที่เขียนขึ้นสามารถเพิ่มประสิทธิภาพในการทำงานในระบบทะเบียนนักศึกษาได้



บทที่ 2 ระบบ LAN

2.1 บทนำ

Microcomputer ได้ออกแบบให้เหมาะสมกับผู้ใช้ (user) คนเดียวและทำงานทีละงาน (Single User Single Tasking) การพัฒนา Microcomputer ได้ก้าวหน้าขึ้นเป็นลำดับขีดความสามารถของ Hardware และ Software สูงขึ้น Microcomputer ถูกพัฒนาเพื่อให้ใช้งานได้ดีขึ้น โดยมีแนวทางในการพัฒนา เพื่อให้สามารถใช้งานได้หลายงานพร้อมกัน (Multi-user multi-tasking) โดยรูปแบบการพัฒนา คือ การนำเอา Microcomputer หลายๆ เครื่องมาต่อรวมกันเป็น Network เพื่อทำงานเป็นระบบ Multi_User โดยแต่ละคนจะมีโมโครโปรเซสเซอร์ของตัวเองที่ทำงานโดยอิสระต่อกัน ระบบ Network ที่ครอบคลุมพื้นที่ทำงานไม่กว้างนักเราใช้ให้ชื่อว่า ระบบ LAN (Local Area Network)

จุดประสงค์ของ LAN

- จุดประสงค์ที่สำคัญของการพัฒนาระบบ LAN ที่พอจะยกมาให้เห็นได้ ดังนี้
- ต้องการแลกเปลี่ยนข้อมูลข่าวสารซึ่งกันและกัน
 - ต้องการใช้งานข้อมูลด้วยความรวดเร็ว สามารถทำการแก้ไข เปลี่ยนแปลงข้อมูลในฐานข้อมูลในที่ต่างๆ ได้อย่างถูกต้อง
 - ต้องการแบ่งการใช้งานอุปกรณ์ต่างๆ เพื่อประโยชน์ในการใช้งานอุปกรณ์ได้คุ้มค่า

ประโยชน์ที่ได้รับจากระบบ LAN

- ความรวดเร็วในการติดต่อสื่อสาร ระบบ LAN ใช้สายข้อมูลเป็นของตนเอง มากกว่าจะใช้สื่อสาธารณะ เช่น สายโทรศัพท์ ทำให้การติดต่อสื่อสารใน Network มีความเร็วสูงช่วยให้ได้ข้อมูลทันต่อเหตุการณ์ ลดความซ้ำซ้อนของข้อมูลและเพิ่มความถูกต้องของข้อมูล
- การติดต่อระหว่างสถานี (workstation) เมื่อสถานีแต่ละตัวเชื่อมต่อเข้าด้วยกันใน Network จะทำให้สามารถติดต่อและส่งข่าวสารหรือข้อมูลถึงกันได้
- การใช้ทรัพยากรร่วมกัน user ที่เข้ามาใช้งานในระบบ จะสามารถใช้งานอุปกรณ์ต่างๆ ร่วมกันได้ เช่น สามารถใช้งาน Hard disk, printer, scanner ร่วมกัน ซึ่งจะช่วยลดการใช้งานซ้ำซ้อนของอุปกรณ์ ทำให้ประหยัดค่าใช้จ่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- การประมวลผลแบบ กระจายงาน (Distributed) เมื่อสถานียมีการใช้งานโปรแกรมและข้อมูลจากศูนย์บริการข้อมูล (file sever) โปรแกรมและข้อมูลจะถูก Load ไปเก็บไว้ที่ Memory ของสถานียนั้นๆ และทำการประมวลผลบนสถานียนั้น เมื่อการประมวลผลเสร็จแล้ว โปรแกรมและข้อมูลเหล่านั้น จะถูกส่งไป เก็บไว้ที่ศูนย์บริการข้อมูลอีกครั้งหนึ่ง เพื่อให้สถานียอื่นสามารถมาเรียกใช้งานได้ต่อไป การประมวลผลแบบกระจายจะช่วยให้ สถานียหลายๆสถานียใช้งานข้อมูลหรือ โปรแกรมร่วมกันได้โดยไม่ไปลดความสามารถในการประมวลผลของแต่ละสถานีย

2.2 องค์ประกอบของ LAN

ระบบ LAN เป็นส่วนหนึ่งของระบบคอมพิวเตอร์ทั่วไปไม่จำกัดอยู่เฉพาะไมโครคอมพิวเตอร์เท่านั้น มินิคอมพิวเตอร์ (mini computer) หรือ เมนเฟรมคอมพิวเตอร์ (mainframe computer) ก็สามารถเชื่อมโยงในลักษณะนี้ได้ แต่ปริมาณการใช้งานไมโครคอมพิวเตอร์จะมีการนำไปใช้มากกว่าจึงเห็นประโยชน์ของการเชื่อมโยงแบบนี้ชัดเจนมากกว่า ดังนั้นการสร้างระบบ LAN ขึ้นมาจึงมีองค์ประกอบไม่ต่างจากคอมพิวเตอร์ในระบบอื่นๆ คือประกอบไปด้วย

- อุปกรณ์ด้านฮาร์ดแวร์ (hardware)
- โปรแกรมควบคุมระบบหรือซอฟต์แวร์ (software)

2.2.1 อุปกรณ์ด้านฮาร์ดแวร์ (hardware)

อุปกรณ์ด้านฮาร์ดแวร์เป็นส่วนประกอบที่สำคัญในระบบ LAN โดยเราแบ่งแยกอุปกรณ์เหล่านี้ออกได้เป็น คือ

- Resource
- Netware Interface Card

Resource หมายถึง อุปกรณ์ต่างๆ ที่มีอยู่ในระบบ เช่น

- Hard disk
- Printer
- Microcomputer

- ตัวกลางนำข้อมูล (Media)

Microcomputer ในระบบ LAN

จากองค์ประกอบของระบบ LAN ประกอบด้วยฮาร์ดแวร์ที่เชื่อมต่อ โปรแกรมควบคุมหรือซอฟต์แวร์ และตัวกลางนำข้อมูล (media) ซึ่งคอมพิวเตอร์ (microcomputer) ที่นำมาเชื่อมต่อกันใน Network ทั้งหมดนั้น จะมีหน้าที่และการทำงานแบ่งเป็น 2 ประเภทคือ

1. เซอร์เวอร์ (server) ในระบบ LAN คือเครื่องคอมพิวเตอร์ทำหน้าที่ให้บริการแก่คอมพิวเตอร์อื่นๆ ที่มาเชื่อมต่อหรือเป็นลูกข่าย โดยทั่วไปหน้าที่ของ server มี 3 ประการ คือ

- บริการในการจัดเก็บข้อมูล เรียกว่า File server
- บริการในการควบคุมเครื่องพิมพ์ เรียกว่า Printer server
- บริการควบคุมอุปกรณ์สื่อสาร เรียกว่า Communication server

ในระบบปฏิบัติการ Netware server จะได้รับการติดตั้งโปรแกรมต่างๆ ในชุดของคำสั่งซึ่งจะมีผลทำให้ server ของ Netware จะทำหน้าที่เป็น File server และ Printer server ในเครื่องเดียวกันแต่ในกรณีของ Communication server นั้นสำหรับ server ของ Netware จะได้รับการติดตั้งให้ทำงานเพียงอย่างเดียว ไม่ซับซ้อน จึงยังไม่นับเป็น Communication server โดยทั่วไป server ทำงานในลักษณะของเครื่องไมโครคอมพิวเตอร์ธรรมดาไม่ได้ ต้องเปิดเครื่องทิ้งไว้เพื่อรอให้บริการแก่คอมพิวเตอร์อื่นๆ เรียกว่า "Dedicated File Server" แต่ในระบบปฏิบัติการบางรุ่นของ Netware ตัว server สามารถทำหน้าที่เป็น server พร้อมกับทำงานเป็นไมโครคอมพิวเตอร์ได้พร้อมๆ กัน เรียกว่า "Non-Dedicated File Server"

2. เวิร์กสเตชัน (Workstation) ในระบบ LAN เครื่องคอมพิวเตอร์อื่นๆ ในระบบที่ user สามารถเข้าไปใช้งานโปรแกรมต่างๆ และติดต่อกับ server ได้เราจะเรียกว่า "สถานีหรือ workstation" ในกรณีของ Non-Dedicated File Server คอมพิวเตอร์ที่เป็น server สามารถทำหน้าที่เป็น workstation ได้พร้อมกัน ตัว workstation จะเป็นไมโครคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการ เช่น DOS, OS/2 เป็นต้น ซึ่งทั้งหมดสามารถเข้าไปใช้ File และเครื่องพิมพ์ของ server ได้ โดยก่อนจะเริ่มติดต่อกับ server ต้องทำการ Load โปรแกรมที่เรียกว่า LAN SHELL ก่อน LAN SHELL ดังกล่าวคือ IPX.exe และ NETX.exe ซึ่งทำงานภายใต้ DOS ถ้า workstation ใช้ OS ตัวอื่น LAN SHELL ก็จะแตกต่างกันออกไป โปรแกรม LAN SHELL จะเกิดขึ้นขั้นตอนที่เรียกว่า Generate shell หรือเรียกย่อๆ ว่า GEN SHELL ซึ่งต้องกำหนดตาม Card Memory และ Interrupt ต่างๆ ของโปรแกรม

shell ให้ตรงกับ Hardware จึงจะเริ่มทำงานได้

ตัวกลางนำข้อมูล (Media)

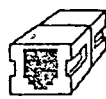
ชนิดของสายส่งข้อมูลที่เชื่อมโยง มีผลอย่างมากต่อราคาในการติดตั้งระบบ ระบบเครือข่ายที่ใช้โพรโทคอลแบบ CSMA/CD ส่วนใหญ่ใช้สายโคแอกเซียล (coaxial) และแบบสายคู่บิดเกลียวหรือสายโทรศัพท์ (twisted pair) สายโคแอกเซียลแบบหนา (thick coaxial) มักใช้กับเครือข่าย Ethernet ที่มีระยะทางการใช้งานไกลกว่าปกติ สายโคแอกเซียลแบบบาง (thin coaxial) จะใช้กับ Ethernet ที่มีระยะทางไม่ไกลนัก ระบบเครือข่ายแบบ Token Passing Bus สามารถใช้สายส่งข้อมูลได้หลายแบบ เช่น ใช้สายโคแอกเซียลแบบ 75 โอห์ม เมื่อต้องการความเร็วในการส่งข้อมูลใช้สายแบบ Semi-rigid (CATV) ก็ได้สำหรับระบบเครือข่ายแบบ Token Ring ควรใช้สายคู่บิดเกลียวแบบหุ้มฉนวน หรือ ที่เรียกว่า Shielded twisted pair (IBM Cabling System type 2) จะดีที่สุดหรืออาจใช้สายแบบไม่หุ้มฉนวน ที่เรียกว่า Unshielded twisted pair (สายโทรศัพท์ หรือ IBM type 3) ได้ในสภาพแวดล้อมที่มีการรบกวนทางไฟฟ้าต่ำ

1. สายส่งข้อมูลชนิดสายคู่บิดเกลียวแบบหุ้มฉนวน (Shielded Twisted Pair) ใช้สายมาตรฐานสองเส้นพันด้วยฉนวนและล้อมรอบด้วยฉนวนชั้นนอก (jacket) อีกทีหนึ่ง สายส่งแบบนี้สามารถส่งข้อมูลได้ด้วยความเร็วสูงถึง 10 Mbps

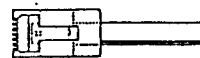
- ข้อดี
- มีราคาถูก
 - ใช้งานง่าย
 - ติดตั้งง่ายและน้ำหนักเบา



bottom view



451A adapter



top view

รูปที่ 2.1 ลักษณะของสายส่งข้อมูลชนิดสายคู่บิดเกลียวแบบหุ้มฉนวน

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น เมื่อผู้ใดได้มาฉบับนี้ควรใช้เพื่อการศึกษาด้านการคำนวณเท่านั้น ไม่ควรนำเอกสารนี้ไปเผยแพร่ในที่สาธารณะ หรือใช้เพื่อการค้าโดยไม่ได้รับอนุญาตจากเจ้าของเอกสาร หากฝ่าฝืนจะถือว่าผิดกฎหมาย และต้องรับผิดชอบต่อเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

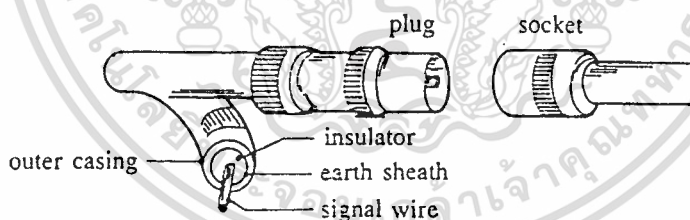
- มีการรบกวนทางไฟฟ้าต่ำ

ข้อเสีย - มีอัตราการส่งข้อมูลต่ำ มีความกว้างแถบสัญญาณ (bandwidth) แคบ ทำให้ไม่เหมาะที่ส่งข้อมูลโดยใช้เทคนิคการส่งข้อมูล (Transmission Techniques) แบบ Analog ที่เรียกว่า บรอดแบนด์

- มีระยะทางการส่งข้อมูลสั้นกว่าสายแบบโคแอกเซียล

2. สายส่งข้อมูลแบบโคแอกเซียล (Coaxial) ประกอบด้วยตัวนำที่ใช้ในการส่งข้อมูลเส้นหนึ่งอยู่ตรงกลางแล้วล้อมรอบด้วยตัวนำอีกตัวหนึ่งทำหน้าที่เป็นสายดิน (ground) ระหว่างตัวนำสองตัวนี้มีฉนวนพลาสติกกันอยู่และทั้งหมดจะล้อมรอบด้วยฉนวนอีกทีหนึ่ง สายส่งข้อมูลแบบนี้สามารถส่งข้อมูลได้ด้วยความเร็วสูงถึง 350 Mbps สายโคแอกเซียลแบบหนา ส่งข้อมูลได้ไกลกว่า สายโคแอกเซียลแบบบาง แต่มีราคาแพงและมีความยืดหยุ่นในการใช้งานน้อยกว่า

ข้อดี - ใช้ได้กับระบบการส่งข้อมูลทั้งแบบดิจิทัล (digital) หรือเบสแบนด์ และแบบอนาล็อก (Analog) หรือ บรอดแบนด์ โดยใช้วิธีการส่งข้อมูล แบบบรอดแบนด์ ที่สามารถใช้ช่วงสัญญาณได้ในช่วงตั้งแต่ 10 ถึง 300 MHz



รูปที่ 2.2 ลักษณะของสายส่งข้อมูลแบบโคแอกเซียล

- สามารถส่งข้อมูลได้ไกลกว่าสายส่งข้อมูลแบบสายคู่บิดเกลียวหุ้มฉนวน
- สามารถส่งได้ในลักษณะที่เป็นทั้ง เสียง ,สัญญาณวิดีโอและข้อมูล
- ติดตั้งง่าย

ข้อเสีย - สายโคแอกเซียลแบบหนามีความแข็ง ทำให้ลำบากในการติดตั้งตามมุม

- มีราคาแพงกว่าสายส่งข้อมูลแบบสายคู่บิดเกลียว หรือ สายโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

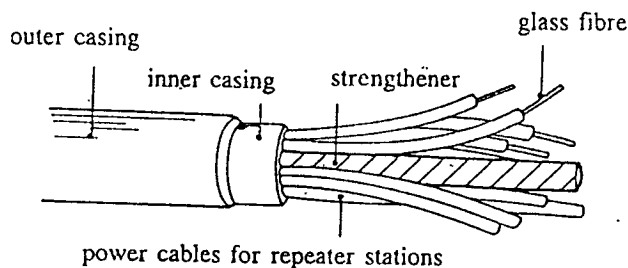
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ค่าใช้จ่ายในการติดตั้งมีราคาสูงกว่าสายส่งข้อมูลแบบสายส่งคู่บิดเกลียว
- เนื่องจากใช้ตัวนำไฟฟ้ามาช่วย ในการหุ้มสายส่งทำให้โอกาสเกิดสัญญาณรบกวน (noise) ได้มาก การเพิ่มตัวหุ้มอีกชั้นหนึ่ง ก็จะช่วยลดปัญหานี้ได้ แต่ค่าใช้จ่ายก็จะเพิ่มขึ้น

3. สายส่งข้อมูลแบบเส้นใยนำแสง (Fiber Optic) ใช้สำหรับการส่งข้อมูลที่ต้องการความเร็วสูงมีข้อมูลที่ส่งจำนวนมากและในสภาพแวดล้อมที่มีสัญญาณไฟฟ้ารบกวนมาก สายส่งข้อมูลแบบนี้จะมีความเร็วในการส่งถึง 1 Gbps หรือมากกว่า

สายส่งข้อมูลแบบเส้นใยนำแสงประกอบด้วยเส้นใย (Fiber) ที่ทำมาจากใยแก้ว 2 ชนิด ชนิดหนึ่งจะอยู่ที่แกนกลาง ส่วนอีกชนิดหนึ่งจะอยู่ด้านนอก ซึ่งใยแก้วทั้งสองนี้จะมี ดัชนีการสะท้อนแสงต่างกัน ทำให้แสงซึ่งถูกส่งออกมาจากปลายหนึ่ง สามารถส่งผ่านไปยังอีกด้านหนึ่งโดยที่ไม่ผ่านออกไปยังพื้นผิวด้านนอกของเส้นใยได้ เส้นใยนี้จะมีการป้องกันด้านนอกอีกชั้นหนึ่ง เพื่อป้องกันการเสียหายภายใน

- ข้อดี**
- ส่งข้อมูลด้วยความเร็วสูง
 - ขนาดเล็กและมีน้ำหนักเบา เส้นใยนำแสงสายเดียว สามารถใช้แทนสายโทรศัพท์ขนาด 300 คู่ได้
 - ไม่มีคุณสมบัติทางไฟฟ้า เพราะใช้แสงในการส่งข้อมูลทำให้สายส่งข้อมูลแบบนี้อาจนำไปใช้ในสภาพแวดล้อมที่ใช้สายแบบอื่นไม่ได้
 - มีการสูญเสียสัญญาณน้อย ทำให้ส่งข้อมูลได้ระยะไกล
 - ไม่สามารถจับสัญญาณได้ การจับสัญญาณจะต้องตัดสายและต่อเข้ากับตัวรับสัญญาณเท่านั้น ซึ่งจะมีประโยชน์สูงในการรักษาความปลอดภัยของข้อมูล



รูปที่ 2.3 ลักษณะของสายส่งข้อมูลแบบเส้นใยนำแสง

ข้อเสีย - มีราคาแพง

- ต้องใช้ความชำนาญในการติดตั้ง
- มีค่าใช้จ่ายในการติดตั้งสูงกว่าสายส่งข้อมูลแบบอื่นๆ
- ส่งข้อมูลได้ทางเดียว ทำให้การติดต่อสื่อสารระหว่างกันต้องใช้สายส่งข้อมูลถึง 2 เส้น ซึ่งจะทำให้ต้องเสียค่าใช้จ่ายเพิ่มขึ้นไปอีก

ปัญหาที่เกิดขึ้นกับสายส่งข้อมูล

ปัญหาสำคัญในการใช้สายส่งข้อมูลก็คือ สัญญาณรบกวน ซึ่งเกิดขึ้นได้ 2 กรณี คือ

1. สัญญาณรบกวนที่เกิดจากสภาพแวดล้อมภายนอก เช่น สายส่งข้อมูลผ่านอุปกรณ์ไฟฟ้าแรงสูง หรืออาจเกิดจากสัญญาณที่แผ่ออกมาจากสายส่งที่อยู่ใกล้กันก็ได้
2. สัญญาณรบกวนที่มาจากตัวสายส่งข้อมูลเอง ไปรบกวนการทำงานของอุปกรณ์อื่นทำให้เกิดปัญหาในการรักษาความปลอดภัยของข้อมูลที่ตามมา การลดการแผ่สัญญาณของสายส่งข้อมูลจะช่วยเพิ่มความปลอดภัยของข้อมูลได้

NETWARE INTERFACE CARD (NIC)

ระบบ Network เริ่มมีความจำเป็นมากขึ้น มีการประมาณกันว่าในอีก 2 ปีข้างหน้า 70% ของ Microcomputer จะเชื่อมต่อเข้ากับระบบ Network การเลือกรูปแบบของ Network ที่เหมาะสมจะมีความสำคัญต่อการขยายระบบในอนาคต ซึ่งจะขึ้นอยู่กับปัจจัยหลายประการ เช่น Hardware, จำนวนสถานีที่ต้องการ, งบประมาณที่มีอยู่ เป็นต้น

NIC ในปัจจุบันจะมีหลายชนิดด้วยกัน ขึ้นอยู่กับ รูปแบบเครือข่าย ที่จะเลือกใช้ว่าต้องการรูปแบบโทน และที่พอจะยกเป็นตัวอย่างได้ดังนี้

Ethernet

ระบบเครือข่ายแบบ Ethernet เป็นการใช้งานระบบเครือข่ายตามมาตรฐาน IEEE 802.3 ที่มีการส่งข้อมูลแบบเบสแบนด์ ความเร็ว 10 Mb/s มีโทโพลยีแบบบัส และใช้โพรโทคอลแบบ CSMA/CD ข้อมูลที่ถูกส่งในระบบเครือข่าย Ethernet จะส่งเป็นกลุ่มข้อมูลสื่อสาร

ของเฟรม ซึ่งรูปแบบของเฟรมมีตามรูปที่ 2.4 ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Preamble	Destn. Address	Source Address	Type Field	Data Field	Frame check Sequence

8 Bytes

6 Bytes

6 Bytes

2 Bytes

46-1500 Bytes

4 Bytes

Ethernet Frame Format

Preamble	Start Delim	Destn Address	Source Address	Type Field	Data Field	Pad Byte	Frame check Sequence

7 Bytes

1 Bytes

2 or

2 or

2 Bytes

0-n Bytes

0-p

4 Bytes

6 Bytes

6 Bytes

Bytes

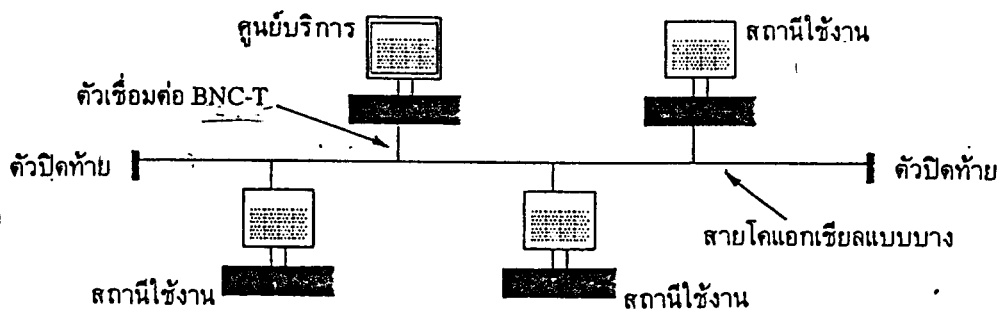
IEEE 802.3 Frame Format

รูปที่ 2.4 รูปแบบกลุ่มข้อมูลสื่อสารของ IEEE 802.3 MAC

เมื่ออุปกรณ์นี้ได้ส่งกลุ่มข้อมูลสื่อสารไปในระบบเครือข่ายแล้ว จะรอนาน้อย 9.6 ไมโครวินาทีก่อนจะส่งข้อมูลอีก ซึ่งช่วงเวลานี้เรียกว่าเป็น Inter-Packet-Gap (IPG) ซึ่ง IPG จะช่วยให้อุปกรณ์รับส่งมีเวลาเตรียมพร้อมที่จะรับกลุ่มข้อมูลสื่อสารโดยที่ไม่เกิดปัญหา ถ้าอุปกรณ์พยายามส่งข้อมูลและพบว่าสายไม่ว่างมันจะรอ ถ้าหลังจากที่ส่งกลุ่มข้อมูลสื่อสารไปแล้วเกิดการชนกันขึ้น อุปกรณ์รับส่งจะรอเป็นช่วงเวลาแบบสุ่ม ก่อนที่เริ่มการส่งกลุ่มข้อมูลสื่อสารที่ชนกันนั้นใหม่ อุปกรณ์รับส่งจะพยายามส่งสูงสุด 16 ครั้ง ถ้ายังส่งไม่ได้จะรายงานว่าเกิดข้อผิดพลาดขึ้น ระบบเครือข่ายแบบ Ethernet ใช้สายส่งข้อมูลแบบโคแอกเซียลได้ 2 แบบ คือแบบบาง (RG-58, 50 โอห์ม) ใช้กับเซกเมนต์ที่มีความยาวน้อยกว่า 185 เมตรและแบบหนา (RG-11, 50 โอห์ม) ซึ่งสามารถติดตั้งในความยาวถึง 500 เมตร ถ้าใช้อุปกรณ์ เช่น ตัวทวนซ้ำสัญญาณ (repeater) ก็สามารถจะเพิ่มความยาวในการใช้งานของสายโคแอกเซียลออกไปได้ถึง 3000 เมตร

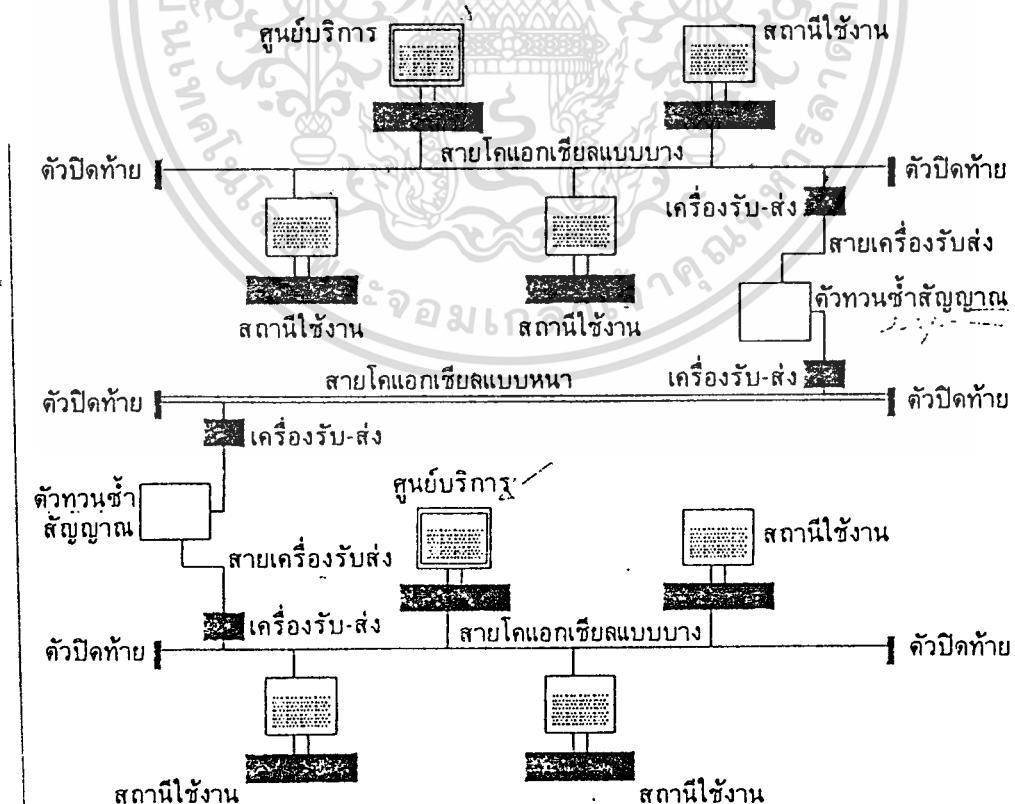
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 การใช้งานระบบ Ethernet โดยใช้สายโคแอกเชียลแบบบาง

สายโคแอกเชียลแบบบางถูกออกแบบมาสำหรับใช้กับระบบเครือข่ายท้องถิ่นภายในสำนักงาน อุปกรณ์ต่างๆ ในระบบจะต้องเข้ากับสายโคแอกเชียลแบบนี้โดยใช้อุปกรณ์ BNC-connector แต่ละสถานีจะต้องห่างกันอย่างน้อย 1 เมตร เพื่อป้องกันไม่ให้สัญญาณรบกวนกันเอง และปลายทางเซกเมนต์จะต้องปิดด้วยตัวเทอร์มิเนเตอร์ BNC แบบ 50 โอห์ม ถ้าใช้สายโคแอกเชียลแบบบางจะต่อสถานีได้สูงสุด 30 สถานี (ดูรูปที่ 2.5 ประกอบ)



รูป 2.6 การใช้งานระบบ Ethernet โดยใช้สายโคแอกเชียลแบบหนา

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่โดยไม่ได้รับอนุญาตด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายโคแอกเซียลแบบหนาหรือที่เรียกว่า backbone Ethernet ใช้งานได้ในระยะไกลกว่าสายโคแอกเซียลแบบบาง และใช้ร่วมกับสายโคแอกเซียลแบบบางได้ด้วยระยะห่างระหว่างแต่ละสถานีต้องไม่เกิน 1 กิโลเมตร เพื่อป้องกันการเกิดการรบกวนกันเองของสัญญาณ สถานีแต่ละสถานีจะต่อกับสายโคแอกเซียลแบบหนา โดยใช้เครื่องรับ-ส่ง (transceiver) ซึ่งระยะห่างระหว่างเครื่องรับ-ส่งแต่ละตัวจะต้องไม่ต่ำกว่า 2.5 เมตรระยะทางจากเครื่องรับ-ส่งถึงสถานีแต่ละสถานีจะต้องโดยใช้สายของเครื่องรับ-ส่งหรือที่เรียกว่า สาย "AUI" (ATTACHMENT UNIT Interface) และต้องยาวไม่เกิน 50 เมตร สายโคแอกเซียลแบบหนาต่อเครื่องรับส่ง ได้สูงสุดถึง 100 ตัว

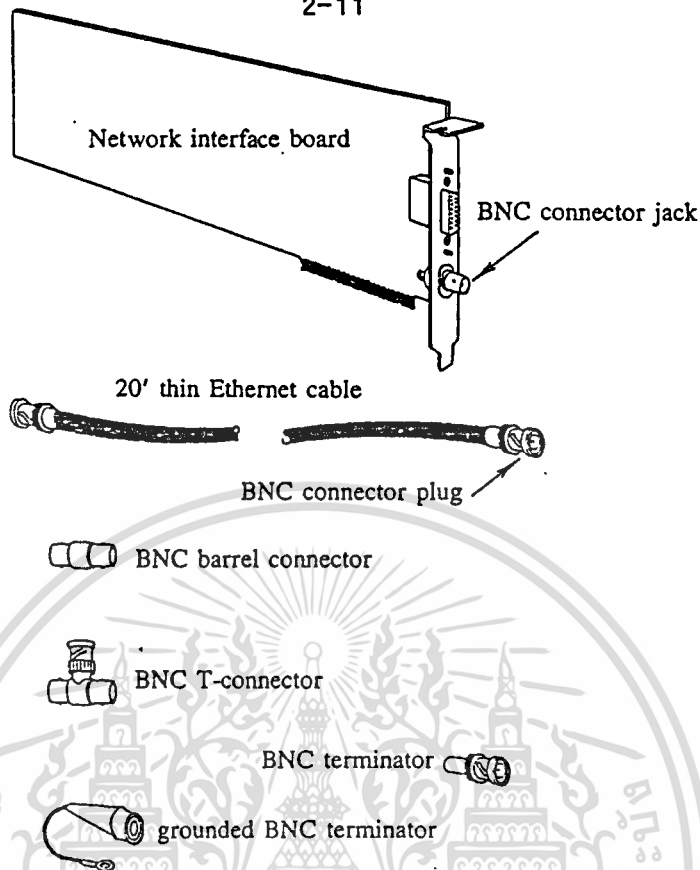
เราสามารถขยายระบบเครือข่ายแบบ Ethernet โดยใช้ตัวทวนซ้ำสัญญาณได้ แต่จำนวนตัวทวนซ้ำสัญญาณระหว่างสองสถานีใด ๆ จะต้องไม่เกิน 2 ตัว ตัวทวนซ้ำสัญญาณจะต่อเข้ากับสายโคแอกเซียลโดยผ่านทางเครื่องรับ-ส่ง และต่อที่จุดใดก็ได้ที่ต่อเครื่องรับ-ส่งได้

การเชื่อมต่อทางกายภาพของ Ethernet

1. ระบบเครือข่าย Ethernet ที่ใช้สายส่งข้อมูลแบบบาง

ฮาร์ดแวร์ที่ใช้ในระบบเครือข่ายที่ใช้สายส่งข้อมูลแบบบางมีดังนี้	
Network Interface Board (NIB)	เป็นแผงวงจรที่สถานีใช้งานแต่ละตัว ที่ต่อกับสายส่งข้อมูล เพื่อส่งข้อมูลในระบบ
BNC Connectors	ตัวเชื่อมต่อระหว่าง NIB กับสายส่งข้อมูล
Thin Ethernet Cable	สาย Ethernet แบบบางเป็นสายส่งข้อมูลแบบ 0.2 นิ้ว RG-58A/U50 โอห์ม
BNC Barrel Connectors	ใช้ต่อสาย Ethernet แบบบางสองเส้นเข้าด้วยกัน
BNC T-Connectors	คล้ายกับ NCB BARREL CONNECTORS โดยที่ปลายสองด้าน ใช้ต่อสาย Ethernet แบบบางสองเส้นเข้าด้วยกันและปลายอีกด้านหนึ่งต่อเข้ากับ NIB
BNC Terminator	เป็นเทอร์มินัลเตอร์แบบ 50 โอห์ม ทำใช้ป้องกันการรบกวนของสัญญาณไฟฟ้า ใช้ที่ปลายทั้งสองด้านของสายส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ฮาร์ดแวร์ที่ใช้ในระบบเครือข่ายที่ใช้สายส่งข้อมูลแบบบาง

2. ระบบเครือข่าย Ethernet ที่ใช้สายส่งข้อมูลแบบหนา

Network Interface Board (NIB) เป็นแผงวงจรที่สถานีใช้งานแต่ละตัวที่ต่อกับสายส่งข้อมูล เพื่อส่งข้อมูลในระบบ

Transceivers Medium Attachment Unit (MAU) สถานีใช้งานในระบบที่ใช้สายส่งข้อมูลแบบหนา จะติดต่อกับระบบเครือข่ายผ่านเครื่องรับส่งที่ต่ออยู่กับสายส่งข้อมูลหลักของระบบ

Transceiver Cable Attached Unit Interface (AUI) Cable เป็นสายคู่บิดเกลียว 4 เส้นหุ้มด้วยฉนวนอีกชั้นหนึ่ง ใช้ในการต่ออุปกรณ์ของระบบเครือข่ายเข้ากับเครื่องรับ-ส่ง

DIX Connector DIX Connector ใช้ต่อระหว่าง NIB กับสายเครื่องรับ-ส่งสัญญาณ

Thick Ethernet Cable สาย Ethernet แบบหนาเป็นสายส่งข้อมูลโคแอกเซียล 50 โอห์ม ขนาดเส้นผ่านศูนย์กลาง 0.4 นิ้ว

N-Series Male Connectors เป็นตัวเชื่อมต่อที่อยู่ด้านปลายสายส่งข้อมูล

N-Series Barrel Connectors ใช้ต่อสาย Ethernet แบบหนา 2 เส้นด้วยกัน

N-Series Terminators

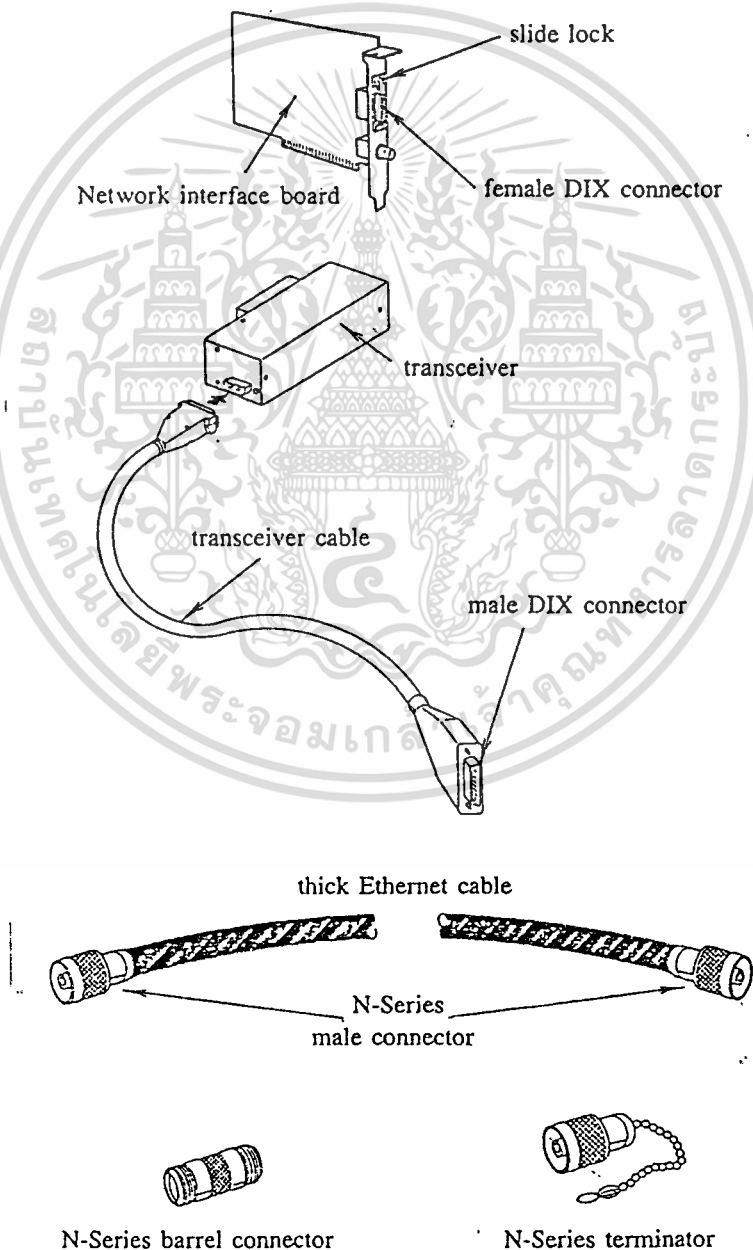
เป็นเทอร์มินเนเตอร์ชนิด 50 โอห์ม ที่ใช้ปิดปลายทั้งสอง
ด้านของสายส่งข้อมูล เพื่อป้องกันสัญญาณรบกวน

Fan-Out Unit

เป็นอุปกรณ์ที่ใช้ต่อกับตัวเครื่องรับ-ส่ง เพื่อช่วยให้ต่อ
อุปกรณ์ของระบบเครือข่ายหลายๆ ตัวเข้ากับเครื่องรับ-
ส่งตัวเดียวได้

Inter-Repeater Link (LRL)

เป็นสายส่งข้อมูลที่ไม่มีอุปกรณ์ของระบบเครือข่ายต่ออยู่



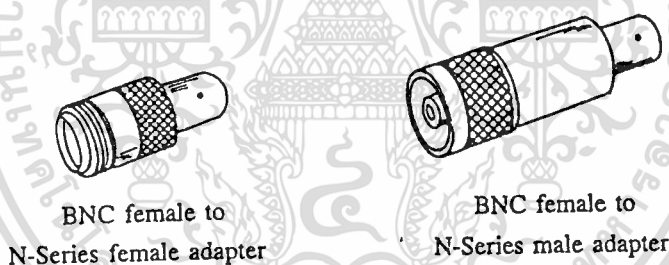
รูปที่ 2.8 ฮาร์ดแวร์ที่ใช้ในระบบ Ethernet ที่ใช้สายส่งข้อมูลแบบหนา

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ห้ามมิให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบเครือข่ายแบบนี้สถานีใช้งานต่อกับตัว เครื่องรับ-ส่งโดยผ่านสาย AUI ซึ่งจะต้องมีความยาวไม่เกิน 50 เมตร และตัวเครื่องรับ-ส่งจะต่อกับสายส่ง ข้อมูลหลักโดยผ่านทาง Physical Medium Attachment

3. ระบบเครือข่าย Ethernet ที่ใช้สายส่งข้อมูลทั้งสองแบบ

ระบบเครือข่ายที่ใช้สายส่งข้อมูลทั้งแบบบางและแบบหนาร่วมกัน ใช้ฮาร์ดแวร์เหมือนกับที่ใช้ในระบบเครือข่ายที่ใช้สายส่งข้อมูลแบบบางและแบบหนา นอกจากนี้ต้องเพิ่มตัวอะแดปเตอร์เพื่อต่อหัวทั้งสองชนิดของทั้ง 2 แบบเข้าด้วยกัน อะแดปเตอร์มี 2 แบบคือ BNC female to N-series female adaptor และ BNC female to N-series male adaptor



รูปที่ 2.9 ลักษณะของ BNC แบบต่างๆ

การใช้สายโคแอกเซียลแบบบางและแบบหนาร่วมกัน ในระบบเครือข่าย Ethernet นี้ ความยาวสูงสุดของแต่ละเซกเมนต์จะอยู่ระหว่างความยาวสูงสุดของสายส่งข้อมูลแบบบางและแบบหนา (185 เมตร และ 500 เมตร ตามลำดับ)

Arcnet

Arcnet เป็นระบบเครือข่ายแบบ token-passing ที่พัฒนาขึ้นโดยบริษัท data point กับระบบคอมพิวเตอร์ ARC Arcnet โดยมีความสัมพันธ์กับมาตรฐาน 802.4 token-passing bus แต่มาตรฐานของ Arcnet จะขึ้นกับบริษัทผู้ผลิตฮาร์ดแวร์ต่างๆ Arcnet ได้กลายมาเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อของระบบเครือข่ายท้องถิ่น ที่ได้รับความนิยมอย่างมากแบบหนึ่ง บอร์ดของ Arcnet จะใช้ไมโครชิพที่ผลิตโดย Standard Microsystem Corporation (SMC) Arcnet พัฒนาขึ้น โดยใช้สายส่งข้อมูลแบบโคแอกเซียล และต่อมาสามารถใช้สายเส้นใยนำแสงในการส่งข้อมูล Arcnet มีระบบปฏิบัติการ สำหรับระบบเครือข่ายท้องถิ่นหลายอัน ที่สนับสนุนเช่น Advance Netware ของ Novell และ VINES ของ Banyan

Arcnet มีข้อดีหลายอย่าง เช่น ง่ายต่อการติดตั้งและบำรุงรักษา และยังเป็นมาตรฐานที่ได้รับการสนับสนุน โดยผู้ผลิตหลายราย ตอนแรก Arcnet ออกแบบมาให้มีการต่อระบบเครือข่ายแบบโมดูลลาร์ โดยใช้ Hub และสายส่งข้อมูลแบบโคแอกเซียล ซึ่งสามารถต่อเครื่องคอมพิวเตอร์ได้สูงสุด 255 เครื่องด้วยกัน ข้อมูลส่งที่อัตรา 2.5 Mb/s ในกลุ่มข้อมูลสื่อสารที่มีขนาดเล็ก 512 ไบต์ โดยใช้โปรโตคอลแบบ token-passing เหมือนกับ token-ring ของ IBM และมีสถาปัตยกรรมแบบดาวกระจาย (Distributed Star) นอกจากนี้ Arcnet ยังปรับตัวมันเองได้โดยอัตโนมัติ ถ้ามีสถานีใดต่อเพิ่มหรือปลดออก ปิดหรือเปิดสถานีใหม่

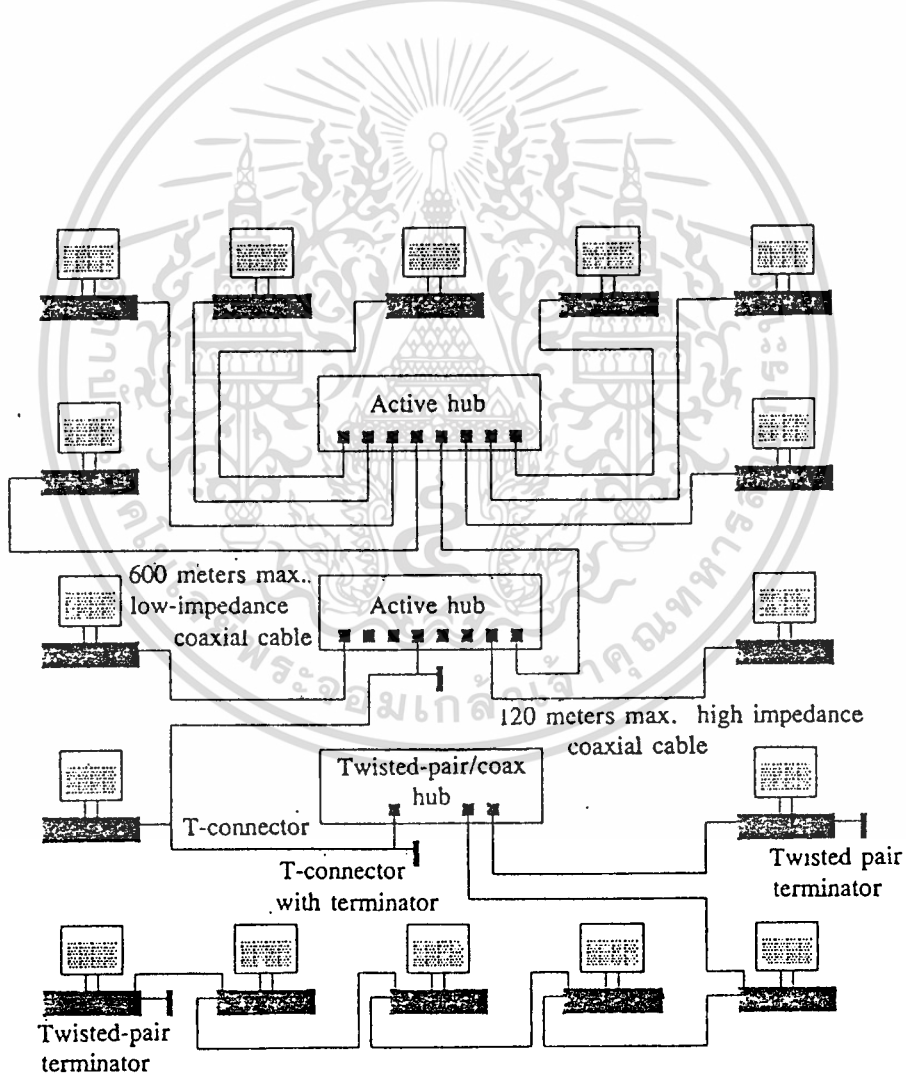
แม้ว่า Arcnet จะมีการส่งข้อมูลที่ช้ากว่า Ethernet ซึ่งมีความเร็วถึง 10 Mb/s หรือระบบเครือข่ายแบบเส้นใยนำแสง ซึ่งมีความเร็วถึง 100 Mb/s แต่ Arcnet ก็ยังคงอยู่ได้ในหลายกรณี Arcnet ก็ทำงานได้ดีกว่าระบบเครือข่ายแบบอื่นอีกด้วย

โครงสร้างของระบบ Arcnet จะเป็นต้นไม้ไม่มีราก (ไม่มีลูบ) แต่ละกิ่งก้านสาขา หรือ เซกเมนต์ของทรี จะต้องอยู่กับฮับ ที่อาจเป็นได้ทั้ง Active hub หรือ Passive hub (ดูรูปที่ 2.10) Active hub เป็นอุปกรณ์ซึ่งต้องมีไฟเลี้ยงที่ใช้ในการสร้างสัญญาณใหม่และขยายสัญญาณ เพื่อช่วยให้ฮับสามารถต่อกับฮับอื่น หรือเครื่องคอมพิวเตอร์อื่นได้ Active hub โดยปกติจะมีตัวเชื่อมต่อ 4-16 ตัว Passive hub เป็นอุปกรณ์ที่ไม่ต้องมีไฟเลี้ยง และจะไม่สร้างสัญญาณใหม่หรือขยายสัญญาณ Passive hub จะจำกัดตัวเชื่อมต่อไม่เกิน 4 ตัว ตัวเชื่อมต่อที่ใช้จะเป็นสายโคแอกเซียลแบบ RG-62 แบบ 93 โอห์ม

ระบบเครือข่าย ARCnet ต้องมีความยาวไม่เกิน 6000 เมตร ระยะสูงสุดระหว่าง active hub และสถานีผู้ใช้หรือ active hub อื่นจะต้องไม่เกิน 600 เมตร ระยะสูงสุดระหว่าง active hub และ passive hub หรือ passive hub กับสถานีผู้ใช้ต้องไม่เกิน 30 เมตร passive hub ไม่สามารถต่อโดยตรงได้และไม่สามารถต่อเป็นลูบได้ สายส่งข้อมูลที่ใช้จะเป็นสายโคแอกเซียล RG-62 แบบ 93 โอห์มโดยใช้ตัวต่อแบบ BNC และเซกเมนต์ที่ต่อกับจุดแบบ

active มีความยาวได้สูงสุด 2000 ฟุต แต่เซกเมนต์ที่ต่อกับจุดแบบ passive ต้องมีความยาวไม่เกิน 100 ฟุต จุดแบบ active สามารถต่อกับ active hub หรืออะแดปเตอร์ของ ARCnet ได้ ทำนั้นตัวเชื่อมต่อบนจุดแบบ active ที่ไม่ได้ใช้งานไม่จำเป็นต้องมีตัวเทอร์มินเตอร์และจุดแบบ passive จะต้องมีระยะทางระหว่างอะแดปเตอร์ต้องไม่น้อยกว่า 1.8 เมตร ทุกเวอร์ชันของ ARCnet จะทำงานความเร็ว 2.5 Mb/s โดยใช้โปรโตคอลแบบเดียวกัน

อะแดปเตอร์ของ ARCnet ใช้ความเร็วในการส่งข้อมูล 2.5 Mb/s โดยใช้โปรโตคอลแบบเดียวกัน ทำให้ระบบเครือข่ายท้องถิ่นแบบ ARCnet สามารถใช้อะแดปเตอร์ และสายส่งข้อมูล



รูป 2.10 การใช้งานระบบเครือข่าย ARCnet

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

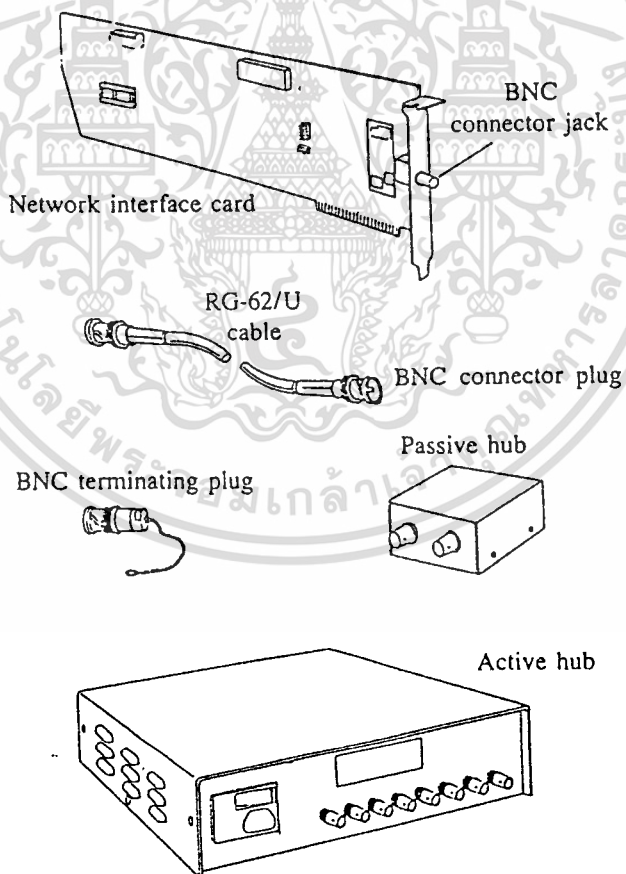
มูลได้หลายแบบ การแปลงระหว่างชนิดของสายส่งข้อมูลจะทำโดยใช้ฮับพิเศษ เช่น SMC two-port twisted-pair, high-impedance coaxial hub

ฮาร์ดแวร์ที่ใช้ในระบบเครือข่าย ARCnet มีดังนี้

Passive Hubs อุปกรณ์ที่ใช้แบ่งสัญญาณของระบบเครือข่าย โดยปกติ 1 ตัวจะมี 4 ช่องทาง ช่องทางที่ไม่ได้ติดต่อกับสถานีใช้งานต้องปิดด้วยเทอร์มิเนเตอร์ชนิด 91 โอห์ม

Active Hubs อุปกรณ์ที่ใช้เพื่อขยายสัญญาณ (condition boost และ relay) ของระบบเครือข่าย โดยปกติ 1 ตัวจะมีอยู่ 8 ช่องทาง

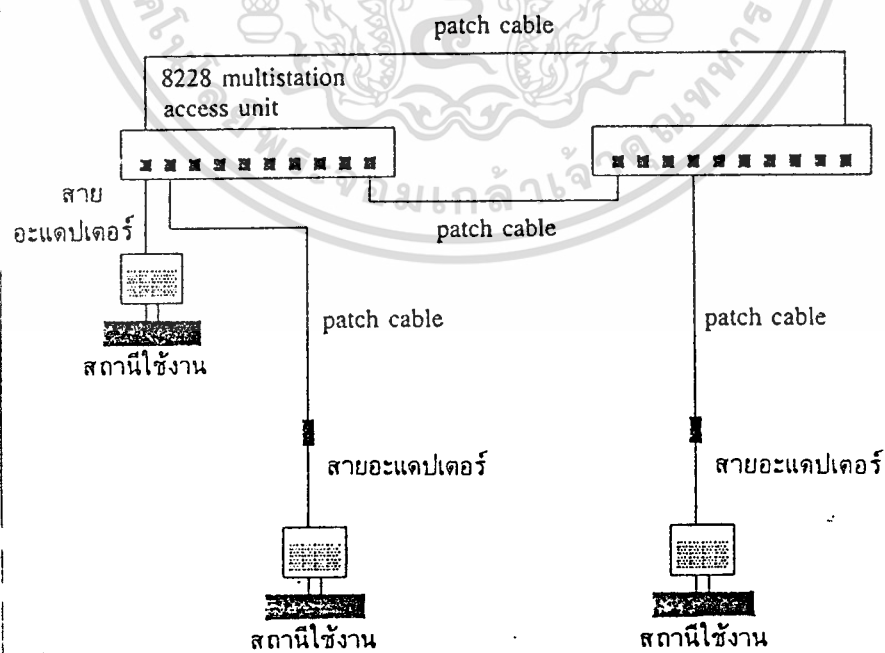
สายส่งข้อมูล ระบบเครือข่ายแบบ ARCnet ใช้สายส่งข้อมูลโคแอกเซียล RG-62/U 93 โอห์ม โดยปลายทั้งสองด้านเป็นหัวเชื่อมต่อ BNC หัวเมีย



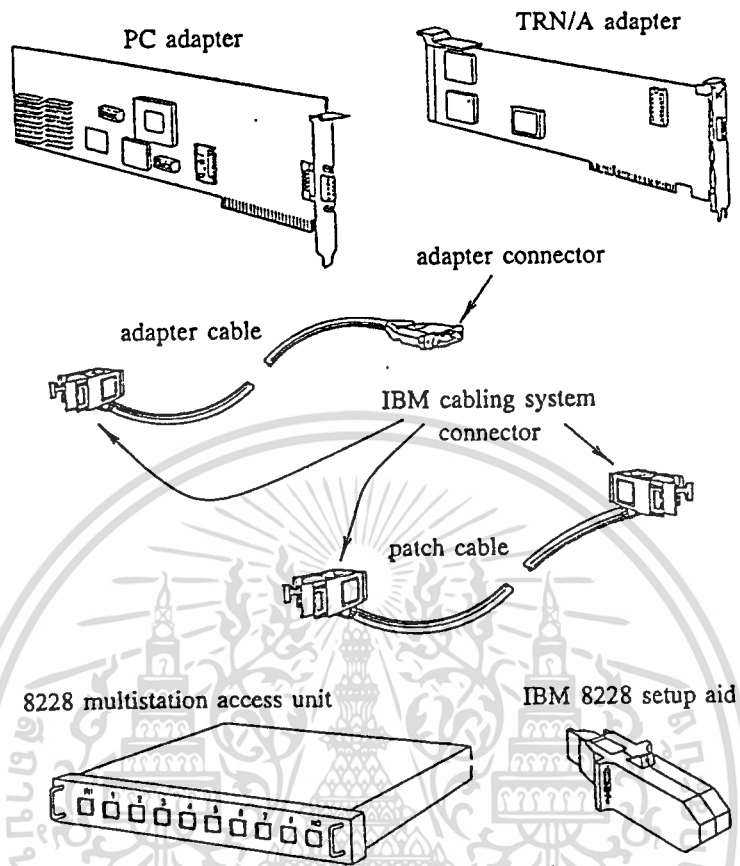
รูปที่ 2.11 ฮาร์ดแวร์ของระบบเครือข่ายแบบ ARCnet

token-ring

ระบบเครือข่ายแบบ Token-Ring เป็นระบบเครือข่ายที่มีความสัมพันธ์กับมาตรฐาน IEEE 802.5 Token-Passing Ring ที่มีโทโพลยีแบบวงแหวนและมีโปรโตคอลแบบ Token-passing ระบบเครือข่ายแบบนี้ ประกอบไปด้วยชุดของสถานีผู้ใช้ที่ต่อกันโดยมีสายส่งข้อมูลเป็นตัวส่งข้อมูลและเป็นการส่งข้อมูลแบบเรียงลำดับ (sequential) จากสถานีหนึ่งไปยังอีกสถานีถัดไป สถานีที่รับจะต้องทวนซ้ำข้อมูลและสร้างสัญญาณใหม่ไปยังสถานีถัดไปในวงแหวน สถานีรับที่มีตำแหน่งตรงกับฟิลต์ของตำแหน่งปลายทางในกลุ่มข้อมูลสื่อสารจะรับข้อมูลนั้นไว้ และประมวลผลตามฟังก์ชัน MAC ของชั้นที่ 2 ใน 802.5 สถานีที่มีตำแหน่งตรงกับตำแหน่งปลายทางจะยังคงส่งข้อมูลนี้ไปยังสถานีถัดไปในวงแหวน สถานีที่เป็นตัวส่งข้อมูล เริ่มแรกจะเป็นตัวลบมันออกจากวงแหวนเอง สถานีแต่ละสถานีจะสามารถส่งข้อมูลได้ก็ต่อเมื่อมันได้รับ token เมื่อสถานีได้รับ token สถานีนั้นจะปรับตัวเพื่อสร้างเฟรมที่ประกอบด้วย start-of-frame sequence ฟิลต์ควบคุมและฟิลต์สถานะ ฟิลต์ตำแหน่ง ฟิลต์ข้อมูล frame check sequence และ end-of-frame sequence หลังจากเฟรมใหม่ถูกสร้างขึ้นมาแล้วสถานีส่งจะ initiate token ใหม่ ซึ่งจะทำ



17560



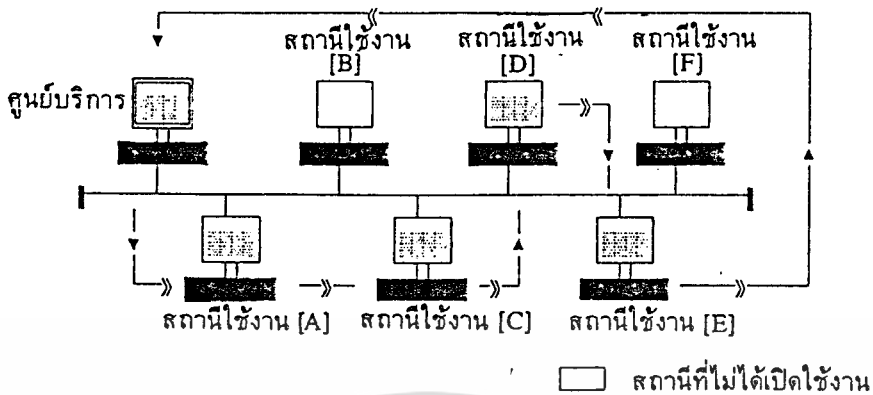
รูปที่ 2.13 ฮาร์ดแวร์ที่ใช้ในระบบเครือข่ายแบบ IBM Token-Ring

ให้สถานีอื่นในวงแหวนใช้งานมันได้ token จะควบคุมเวลาและจำนวนเวลาสูงสุดที่ สถานีจะสามารถใช้ตัวกลางได้ ทั้งการส่งเฟรมใหม่หรือไม่ส่ง ก่อนที่จะมีการส่ง token ออกไป ระบบเครือข่ายแบบ token-ring สามารถใช้สายส่งได้ 2 แบบ คือ แบบ shielded twisted pair (type 1,2 และ 9) และแบบ unshielded telephonestyle wire (type 3)

token-bus

เป็นระบบเครือข่ายที่ใช้งานตามมาตรฐาน IEEE 802.4 โดยกำหนด token bus ที่สามารถนำไปใช้งานโดยใช้ทั้งแบบเบสแบนด์ และ บรอดแบนด์ หรือ hybrid application ได้ บัสจะมีการทำงานคล้ายกับ token-ring ตรงที่อัลกอริทึมในการส่ง token จะเป็นแบบเรียงลำดับจากสถานีหนึ่ง ไปยังอีกสถานีหนึ่ง token bus จะมีการทำงานเชิงตรรกเป็นระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.14 การทำงานของระบบเครือข่ายแบบ Token-Passing Bus

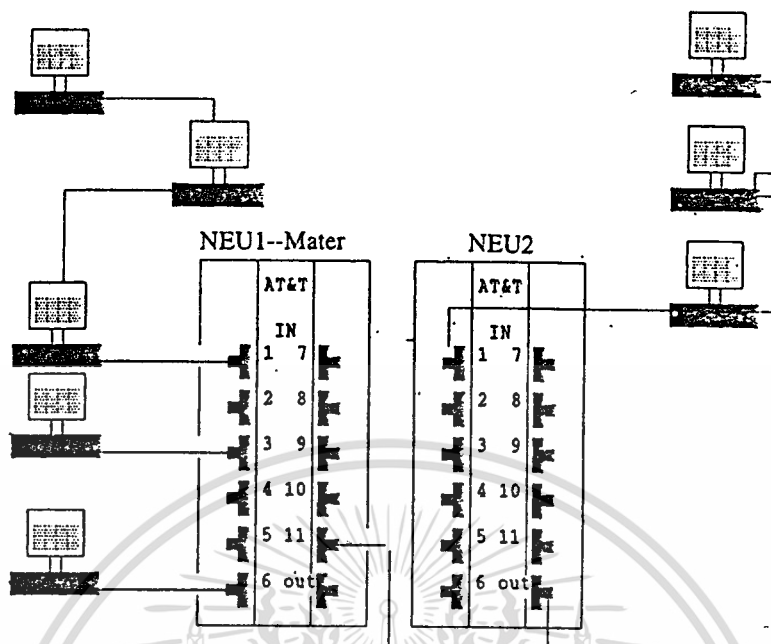
หลังจากที่สถานีได้ส่งเฟรมของข้อมูลใดๆ เรียบร้อยแล้วสถานีจะส่งเฟรม token MAC ไปให้ตัวต่อจากมัน แล้วคอยฟังการตอบรับที่ตัวต่อจากมันว่าได้รับเฟรมที่ส่งไปให้หรือไม่ แล้วตัวนั้นก็ จะทำงาน (active) ถ้าผู้ส่ง token ไม่ได้รับเฟรมที่ถูกต้องเมื่อคอยฟังครบ 4 ครั้งแล้ว มันจะ ส่งข้อมูลใหม่อีกครั้ง แล้วถ้าตัวต่อไปจากมันยังไม่ส่งเฟรมที่ถูกต้องมาอีก โดยจะถือว่าตัวต่อไปจาก มันเสีย ผู้ส่งจะส่งเฟรม who follows ที่จะไปถามว่าใครเป็นเป็นสถานีที่อยู่ถัดไป เพื่อช่วยให้ผู้ ส่งสามารถกำหนดสถานีถัดไปที่จะส่งข้อมูลไปให้ และส่ง token ไปให้

starLAN

เป็นการใช้งานระบบเครือข่ายที่ใช้สายคู่บิดเกลียวของ AT&T โดยใช้โปรโตคอลแบบ CSMA/CD มีเร็วในการใช้งานระบบน้อยกว่าแบบ Ethernet โดยแบบ starLAN มีความเร็ว 1 Mb/s ส่วนของ Ethernet มีความเร็วถึง 10 Mb/s แต่แบบ starLAN จะเดินสายส่งได้ ง่ายกว่า เมื่อเทียบกับสายโคแอกเซียล

ระบบเครือข่ายท้องถิ่นแบบ starLAN ใช้โทโพโลยีแบบดาว, daisy chain หรือทั้งสอง แบบร่วมกันได้ ในการต่อแบบ daisy chain จะมีข้อดีที่อุปกรณ์แต่ละตัวจะเชื่อมต่อกับตัวต่อไป แบบอนุกรมเชื่อมต่อแบบตัวที่ ซึ่งเป็นตัวให้สัญญาณออกของสถานีหนึ่งจะเป็นสัญญาณเข้าของอีกสถานี หนึ่งดังรูป (ดูรูปประกอบ 2.15) การต่อแบบ daisy chain เหมาะสำหรับระบบเครือข่ายที่มี ไม่เกิน 10 สถานี และระยะห่างระหว่างปลายทั้งสองด้านต้องไม่เกินกว่า 122 เมตร ซึ่งถ้าเกิน กว่านี้ จะต้องใช้โทโพโลยีแบบดาวแทน ซึ่งจะเป็นการต่อแต่ละอุปกรณ์ออกจากศูนย์กลางที่ เรียกว่า

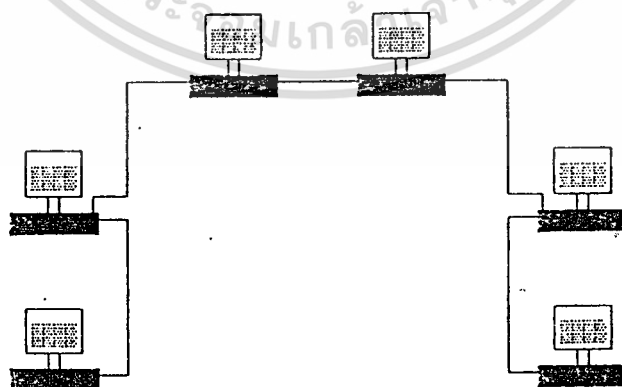
Network Extension Unit (NEU) เพราะ NEU แต่ละตัวจะต่อสถานีออกมาได้สูงสุด 11 สถานี ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.15 การใช้ระบบเครือข่าย starLAN

และสายส่งข้อมูลของแต่ละสถานีจาก NEU จะยาวได้สูงสุด 245 เมตร ถ้าระบบมีสถานีมากกว่า 11 สถานีก็ต้องใช้ NEU หลายตัว ซึ่งจะมี NEU ตัวหนึ่งเป็นหลักให้ NEU ตัวอื่นมาต่อเข้ากับ NEU หลักด้วยสายส่งที่จะต้องมีความยาวไม่เกิน 3 เมตร

จะเห็นว่า starLAN มีความยืดหยุ่นที่ยอมให้ทั้ง daisy chain และโทโพโลยีแบบดาว ต่ออยู่ในระบบเครือข่ายเดียวกันได้ ดังรูป (ดูรูป 2.16) แสดงการต่อแบบ daisy chain ที่มี



รูป 2.16 การใช้ระบบเครือข่ายแบบ starLAN ขนาดเล็กโดยใช้ daisy chain

2 ตัว ถึง 10 สถานีสามารถต่อออกจากพอร์ตของ NEU ตัวเดียวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 โปรแกรมควบคุมระบบหรือซอฟต์แวร์

การที่ระบบจะรู้ว่าข้อมูลที่ใช้งานอยู่ในเครื่องคอมพิวเตอร์ของตนเองหรืออยู่ในเครื่องอื่นในระบบ โปรแกรมควบคุมระบบ Network (Network Operating System) จะเป็นตัวจัดการ โปรแกรมควบคุมระบบ LAN แบ่งการทำงานเป็น 2 ประเภทคือ

- ทำงานภายใต้ระบบปฏิบัติการอื่น เช่น DOS, OS/2
- สร้างระบบปฏิบัติการขึ้นมาเอง เช่น Netware, VINES

ระบบปฏิบัติการ Netware

Netware เป็นผลิตภัณฑ์ของ Novell มีการใช้งานกันอย่างกว้างขวาง เป็นมาตรฐานอุตสาหกรรมและเป็นผู้นำในด้านซอฟต์แวร์ ฮาร์ดแวร์ และมาตรฐานของระบบ LAN ในปัจจุบัน Netware มีข้อดีคือ

- ความเข้ากันได้ Netware สามารถจะใช้งานร่วมกับซอฟต์แวร์และฮาร์ดแวร์ที่ไม่ใช่ Netware บนระบบ Netware เดียวกัน เช่น โปรแกรมใน Netware สามารถใช้งานร่วมกับ PC DOS ได้

- ฮาร์ดแวร์ของ Netware Netware เป็นระบบ Network ที่ไม่ขึ้นอยู่กับฮาร์ดแวร์ (Hardware-Independent) Netware สามารถใช้งานกับฮาร์ดแวร์มากกว่า 30 ชนิด และสนับสนุน Topology ทุกชนิดในระบบเครือข่าย ไม่ว่าจะเป็นระบบเครือข่ายแบบ Token-bus StarLAN, Ethernet Netware ติดต่อกับ User โดยไม่สนว่าจะใช้ฮาร์ดแวร์อะไรก็ตาม

- ความสามารถในการเชื่อมต่อ Netware สามารถเชื่อมต่อหรือมีสะพานสื่อสารกับระบบเครือข่ายทุกชนิดที่ Netware สนับสนุนอยู่ การติดต่อระหว่างระบบเครือข่ายที่เชื่อมสะพานสื่อสารเข้าด้วยกันและใช้งานข้อมูลร่วมกัน จะแบบไม่สนใจฮาร์ดแวร์ โดยผู้ใช้สามารถทำงานได้ตามปกติ

- ความสามารถในการขยายระบบ ระบบ Network ที่ใช้ Netware สามารถขยายเนื้อที่เก็บข้อมูลและหน่วยความจำ ซึ่งสามารถเพิ่มเข้าไปที่ศูนย์บริการข้อมูล (server), การขยายระบบทำได้โดยการเพิ่มสถานีของผู้ใช้ หรือ เพิ่มศูนย์บริการข้อมูล (server) แต่จะต้องคำนึงถึงรูปแบบของเครือข่ายด้วย, การเพิ่มประสิทธิภาพของระบบ ทำได้เช่น การเพิ่มประสิทธิภาพการทำงานของศูนย์บริการข้อมูล เป็นต้น

ระบบปฏิบัติการ NETWARE

ความเป็นมาของ Netware

เดิมทีนั้น Novell พัฒนา Netware ขึ้นมาเพื่อเป็นระบบปฏิบัติการสำหรับแม่ข่ายของระบบเครือข่ายที่ Novell ตั้งชื่อว่า ระบบ S-Net ซึ่งเป็นระบบเครือข่ายที่ใช้รูปแบบการต่อแบบดาว (star) มีแม่ข่ายที่ใช้ cpu MC68000 ของโมโตโรลาเป็นศูนย์กลาง ต่อมาเมื่อเครื่องไมโครคอมพิวเตอร์ IBM PC XT กลายเป็นที่นิยมแพร่หลายมากขึ้น หลังจากประกาศตัวในปี 1981 เพื่อเป็นการสนองตอบความต้องการในการเชื่อมต่อคอมพิวเตอร์เข้าด้วยกันเป็น Network Novell จึงหันมาพัฒนา Network บนเครื่องไมโครคอมพิวเตอร์ที่ใช้สถาปัตยกรรมของ INTEL 8088 แทน

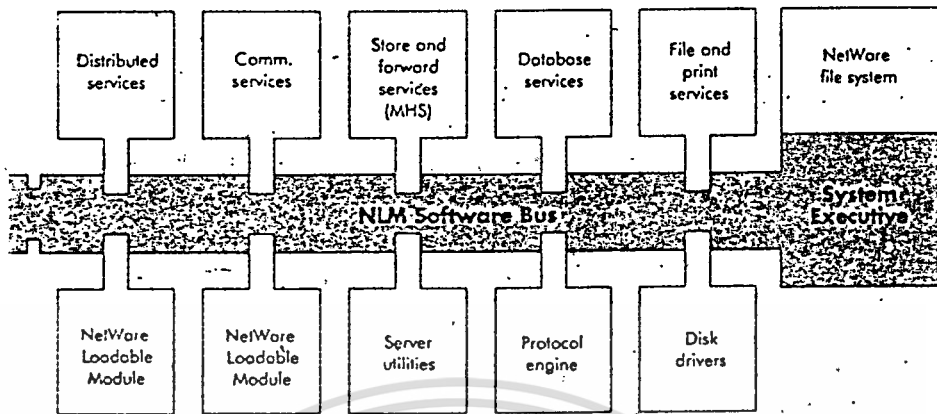
Netware เวอร์ชันแรกถูกพัฒนาออกมาเมื่อต้นปี 1983 ตามมาด้วย Advanced Netware V1.0 ในปี 1985 และในปีเดียวกันนี้ Novell ก็ได้ประกาศตัว Advanced Netware V2.1 ซึ่งถือเป็นระบบปฏิบัติการตัวแรกทำงานอยู่ใน PROTECTED MODE ของ INTEL 80286 PC

ต่อจากนั้น Advance Netware V2.0 ก็ถูกพัฒนาออกมาในปี 1986 เพื่อใช้ความสามารถเต็มที่ของ 80286 PROTECTED MODE ทำให้ Netware สามารถใช้หน่วยความจำมากกว่า 640 KB ได้ นอกจากนั้น Netware เวอร์ชันนี้เป็นเวอร์ชันแรกที่สามารถเชื่อมต่อบน Network ได้ถึง 4 ระบบ เข้าด้วยกันที่เครื่องแม่ข่าย

ในปี 1987 Novell พัฒนา SFT Netware ออกมาเพื่อเพิ่มความสามารถของ Netware ในหลายๆ ด้าน เช่น ด้านการจัดการและการบริหารระบบ ความสามารถที่จะมีฮาร์ดดิสก์ 2 ตัวที่แม่ข่าย เพื่อให้ฮาร์ดดิสก์ตัวที่สองทำหน้าที่สำรองข้อมูลของฮาร์ดดิสก์ตัวแรกโดยอัตโนมัติ (Disk Mirroring or Disk Duplexing) และเปิดโอกาสให้ผู้ใช้งานหรือผู้พัฒนาซอฟต์แวร์อื่นๆ สามารถที่จะพัฒนาโปรแกรมสำหรับทำงานที่เครื่องแม่ข่ายในลักษณะของ Value-added Process (VAP)

Netware V2.15 และ Netware สำหรับแมคอินทอช ถูกพัฒนาออกสู่ตลาดในเดือนธันวาคม 1988 เพื่อรองรับการเชื่อมระบบ Netware ของเครื่องแอมแปปเปิลแมคอินทอชกับระบบเครือข่ายไมโครคอมพิวเตอร์ของ Netware

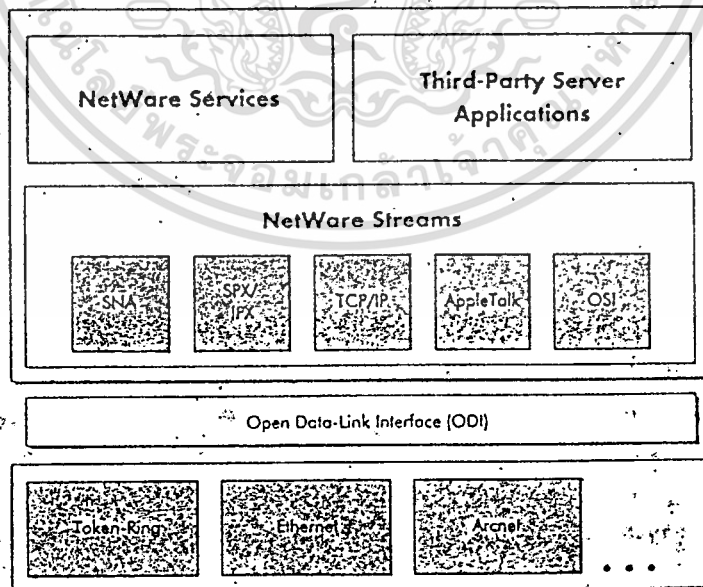
เดือนกันยายน 1989 Novell ประกาศตัว Netware 386 V3.0 ซึ่งเป็นระบบปฏิบัติการแบบ 32 bit สำหรับไมโครโปรเซสเซอร์ 80386 และ 80486 Netware 386 ได้พัฒนาขึ้นอีกขั้นหนึ่ง ในเรื่องของประสิทธิภาพ และ ยึดความสามารถในการทำงานระบบรักษาความปลอดภัย ความคล่องตัวในการทำงาน การรองรับโปรโตคอลต่างๆ สำหรับการสื่อสารและการรองรับการ



รูปที่ 2.17 ปรัชญาการออกแบบ Netware V3.11

ทำงานในลักษณะของ distributed application

หลังจากที่ Netware 386 V3.1 ออกสู่ตลาดใน เดือนมิถุนายน 1990 เมื่อต้นปี 1991 Novell ก็ได้นำ Netware เวอร์ชันล่าสุดออกมา คือ Netware V2.2 และ Netware V3.11



รูปที่ 2.18 โพรโตคอลของ Netware V3.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Netware V2.2 เป็นระบบปฏิบัติการที่รวมเอาความสามารถของ Netware 286 รุ่นเดิมของ Novell คือ SFT, Advance และ ELS Netware เข้ามาไว้ด้วยกัน

ส่วน Netware V3.11 เป็นเวอร์ชันที่พัฒนาขึ้นมาจาก Netware 386 V3.10 โดยเพิ่มความสามารถในการเชื่อมต่อกับระบบ UNIX ผ่านโปรโตคอล TCP/IP การเชื่อมต่อกับระบบเครือข่ายของแอปเปิลและการจัดการระบบจากศูนย์กลางซึ่งอยู่ห่างไกล (Remote Management Facility)

2.2.3 ปัจจัยที่ทำให้ Netware เป็นที่นิยม

เหตุผลที่สำคัญมีอยู่ 4 ประการ คือ
ประสิทธิภาพภายในการทำงาน

Netware เป็นระบบปฏิบัติการ ที่ถูกพัฒนาขึ้นมาเพื่อให้ทำงาน เป็นแม่ข่ายโดยเฉพาะ สถาปัตยกรรมของเน็ตแวร์ที่ถูกออกแบบมาโดยเฉพาะ จึงทำให้เน็ตแวร์ทำงานได้เร็วและมีประสิทธิภาพกว่าระบบปฏิบัติการอื่นๆ ที่ถูกพัฒนาขึ้นมาให้ทำงานอยู่บนระบบปฏิบัติการของ DOS, UNIX หรือ OS/2 เทคโนโลยีต่างๆ ที่ โนเวลล์นำมาผนวกไว้ในเน็ตแวร์ จึงทำให้เน็ตแวร์เป็นผู้ที่ได้ประสิทธิภาพการทำงานในหลายลักษณะ ได้แก่

- Multitasking Kernel Netware เป็นระบบปฏิบัติการที่สามารถทำงานหลายงานขนานกันไป นอกจากนี้ยังมีการทำงานเป็น multitreaded ด้วย จึงทำให้แม่ข่ายของ Netware สามารถตอบสนองความต้องการของผู้ใช้หลายๆ คนได้พร้อมกัน และสามารถทำงานได้อย่างดีภายใต้การใช้งานอย่างหนัก

- Elevator Seeking ส่วนหนึ่งของ Netware จะทำหน้าที่เกี่ยวกับการค้นข้อมูลจากฮาร์ดดิสก์ของแม่ข่าย(Disk read process) Read request จากลูกข่ายจะถูกจัดเรียงลำดับของการอ่าน และจัดเรียงลำดับความสำคัญให้ขึ้นอยู่กับตำแหน่งและทิศทาง การเคลื่อนไหวของหัวอ่านฮาร์ดดิสก์ในขณะนั้น เทคนิคนี้จะทำให้มีการเคลื่อนไหวมาของหัวอ่านฮาร์ดดิสก์ในขณะนั้น เทคนิคนี้จะทำให้มีการเคลื่อนไหวมาของหัวอ่านน้อยที่สุด ซึ่งทำให้ฮาร์ดดิสก์มี throughput ที่สูงกว่าระบบที่ไม่ใช้ เทคนิคนี้ภายใต้ภาวะการทำงานอย่างหนัก

- Disk Caching จะเก็บข้อมูลของไฟล์ที่มีการเรียกใช้งานบ่อยๆ ไว้ในหน่วยความจำ (cache memory) เพราะการอ่านจากหน่วยความจำจะทำงานได้เร็วกว่าการอ่านจากฮาร์ดดิสก์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้า ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการอ่านข้อมูลจากฮาร์ดดิสก์แต่ละครั้ง Netware จะอ่านข้อมูลจากฮาร์ดดิสก์มากกว่าความต้องการจริง แล้วเอามาเก็บไว้ที่หน่วยความจำเพื่อเตรียมไว้ล่วงหน้าสำหรับการขอข้อมูลในอนาคต วิธีนี้ช่วยลดจำนวนครั้งที่จะต้องมีการอ่านจริงจากตัวฮาร์ดดิสก์ ซึ่งทำให้มีการทำงานรวดเร็วยิ่งขึ้น

- **Background Writes** อีกส่วนหนึ่งของ Netware จะทำหน้าที่เกี่ยวกับการเขียนข้อมูลลงบนฮาร์ดดิสก์ของแม่ข่าย (disk write process) การแยกขบวนการในการเขียนและขบวนการในการอ่านข้อมูลจากฮาร์ดดิสก์ออกจากกัน ทำให้การเขียนข้อมูลสามารถทำงานในลักษณะของ background task ซึ่งสามารถจึงทำในระหว่างช่วงที่มีการใช้งานน้อยลง

- **Indexed File Allocation Table (Turbo FATS)** ระบบไฟล์ของ Netware ใช้ File allocation tables (FATS) ในการจัดเก็บและค้นหาข้อมูลของไฟล์ในฮาร์ดดิสก์ ถ้าหากไฟล์ใดมีขนาดใหญ่และใช้เนื้อที่เกิน 64 บล็อก Netware จะทำ index ของ FAT สำหรับไฟล์นั้น เพื่อช่วยให้การค้นหาข้อมูลสามารถทำได้รวดเร็วยิ่งขึ้น

ความน่าเชื่อถือของระบบ

Netware มีคุณสมบัติพิเศษหลายๆ อย่างสำหรับป้องกันความเสียหาย หรือ ความไม่สมบูรณ์ของข้อมูลการที่ Netware เป็นผู้นำของตลาดในด้านนี้ก็เป็นที่จุดหนึ่งที่ทำให้ Netware สามารถครองความเป็นหนึ่งในตลาดในปัจจุบันซึ่ง Netware มีข้อดีเด่นหลายประการดังนี้

- **Read-after-write verification** ทุกครั้งที่มีการบันทึกข้อมูล ลงบนฮาร์ดดิสก์ Netware จะทดลองอ่านข้อมูลกลับมาอีกครั้ง เพื่อตรวจสอบว่าข้อมูลนั้นสามารถอ่านกลับมามีอีก

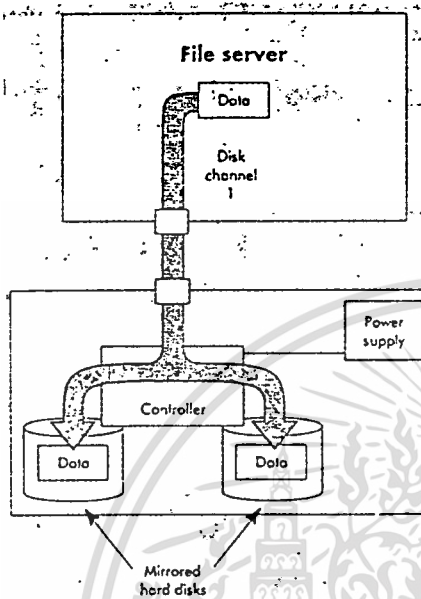
- **Duplication of directories** Netware จะเก็บข้อมูลเกี่ยวกับโครงสร้างของไดเรกทอรีไว้สองที่ ถ้าหากข้อมูลของโครงสร้างไดเรกทอรีเกิดเสียหาย Netware จะใช้ข้อมูลจากชุดสำรองแทนในการเรียกหาข้อมูลจากฮาร์ดดิสก์

- **Duplication of file allocation tables** Netware จะมีการเก็บข้อมูลเกี่ยวกับ file allocation table ไว้สองที่ เพื่อป้องกันกรณีที่ FAT เกิดเสียหาย

- **Hot fix** ขณะที่กำลังมีการเขียนข้อมูลลงบนฮาร์ดดิสก์ ถ้าหาก Netware พบว่า เนื้อที่ที่จะเขียนข้อมูลลงไปในนั้นเกิดเสียหาย Netware จะย้ายไปเขียนลงที่ใหม่แทนโดยอัตโนมัติ พร้อมทั้งจะทำการบันทึกตำแหน่งของฮาร์ดดิสก์ที่เสียหายไว้

เอกสารนี้เป็นเอกสารของ **System Fault Tolerance (SFT)** ในการป้องกันความเสียหายหรือความไม่สมบูรณ์ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

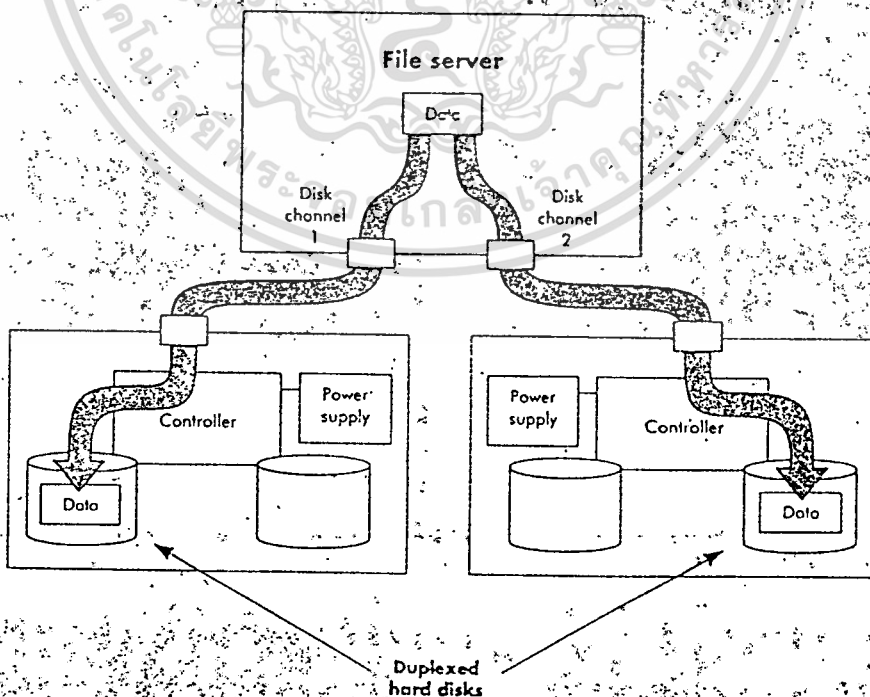
ของข้อมูล เราสามารถเลือกใช้ความสามารถของ Netware ในการทำ Disk mirroring, Disk Duplexing หรือ Transaction Tracking System (TTS) ได้



รูปที่ 2.19 ลักษณะของการเข้าถึงข้อมูล

ในแบบ disk mirroring

- Disk Mirroring คือ การติดตั้งฮาร์ดดิสก์เป็นคู่ เพื่อให้ฮาร์ดดิสก์ตัวหนึ่ง ทำหน้าที่สำรองข้อมูลของฮาร์ดดิสก์อีกตัวหนึ่ง โดยอัตโนมัติ



รูปที่ 2.20 ลักษณะการเข้าถึงข้อมูลในแบบ disk duplexing

เอกสารนี้เป็นเอกสารที่... ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Disk Duplexing มีลักษณะคล้าย disk mirroring แต่เป็นการสำรองทั้งฮาร์ดดิสก์ และ ฮาร์ดดิสก์คอนโทรลเลอร์
- Transaction Tracking System ในการบันทึกข้อมูลแต่ละครั้ง ซึ่งประกอบไปด้วย ข้อมูลของหลายๆ ไฟล์ หลายๆ สำเนาที่เราเรียกว่าการทำ Transaction นั้น ถ้าข้อมูลส่วนหนึ่งส่วนใดไม่ได้ถูกบันทึก โดยสมบูรณ์ทั้งหมด, ฐานข้อมูลของเราก็จะเป็นฐานข้อมูลที่ไม่ถูกต้อง ถ้าหากมีการใช้ TTS Netware ก็จะช่วยตรวจสอบว่าข้อมูลที่เกี่ยวข้องในแต่ละ Transaction ได้มีการบันทึกจนครบถ้วนหรือไม่ ถ้าหากมีเหตุขัดข้องที่ทำให้การบันทึกข้อมูลบางส่วนไม่เรียบร้อย Netware จะสามารถยกเลิก Transaction นั้น และเรียกข้อมูลเดิมกลับมาให้อยู่สภาวะสมบูรณ์เหมือนเดิม ก่อนที่จะมีการทำ Transaction นั้นๆ

ระบบรักษาความปลอดภัย

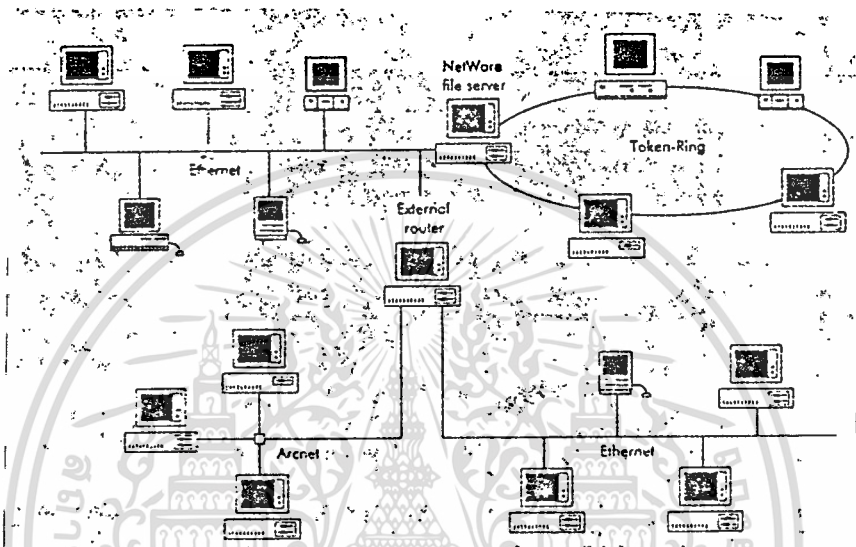
ส่วนของ Netware ที่เกี่ยวข้องกับความปลอดภัยของระบบ เป็นส่วนประกอบขั้นพื้นฐานที่ทำให้ระบบเครือข่ายของ Netware มีระบบรักษาความปลอดภัยที่ไม่ด้อยกว่า ระบบมินิคอมพิวเตอร์ หรือ ระบบเมนเฟรมคอมพิวเตอร์ เลย การที่ระบบไฟล์ของ Netware เป็นระบบที่พัฒนาขึ้นมาโดยเฉพาะ ทำให้ผู้ใช้ไม่สามารถเรียกค้นหาไฟล์ของ Netware ผ่าน DOS, UNIX หรือระบบปฏิบัติการอื่นได้ ถึงแม้ว่าจะสามารถเข้าไปทำงานอยู่ที่เครื่องแม่ข่ายโดยตรง

ระบบรักษาความปลอดภัยของ Netware ประกอบไปด้วยระดับต่างๆ ดังนี้

- Account Security
- Password Security
- Directory Security
- File Security
- Internetwork Security

Novell เป็นผู้ริเริ่มการนำเอาชื่อผู้ใช้ (Username) รหัสผ่านประจำตัวผู้ใช้ (Password) และรายละเอียดเกี่ยวกับผู้ใช้ (User Profile) มาใช้กับระบบเครือข่ายไมโครคอมพิวเตอร์ ซึ่ง User Profile ของผู้ใช้แต่ละคนจะประกอบไปด้วยรายชื่อและสิทธิในการใช้งานเอกสารงานของทรัพยากรต่างๆ ที่มีอยู่ในระบบโดยที่ผู้ดูแลระบบสามารถที่จะระบุใน เวลา และสถานที่ ที่ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้แต่ละคนจะสามารถติดต่อเข้ามาใช้งานได้ Netware จะสามารถตรวจสอบและป้องกันการบุกรุก



รูปที่ 2.21 การเชื่อมโยงระบบเครือข่ายในลักษณะ external router

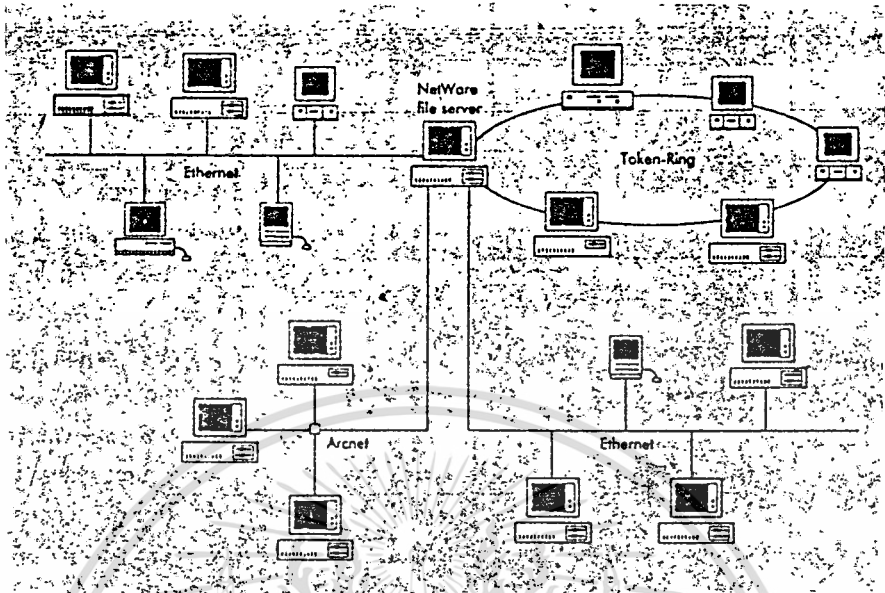
รุกรเข้ามาใช้งานโดยไม่ได้รับอนุญาต (intruder detection and lockout) พร้อมทั้งจะมีการแจ้งให้ผู้ดูแลระบบทราบถึงการบุกรุกได้ด้วย

รหัสผ่านประจำตัวผู้ใช้ ผู้ใช้จะถูก Encrypt ก่อนเก็บไว้ที่ฮาร์ดดิสก์ของแม่ข่าย พร้อมกันนั้นผู้ดูแลระบบสามารถบังคับให้ผู้ใช้ต้องเปลี่ยนรหัสผ่านประจำตัวทุกๆ ระยะเวลาที่กำหนดไว้ได้

กุญแจสำหรับถอดรหัส (Password keys) จะถูก Encrypt ก่อนถูกส่งผ่านสายเคเบิลของเครือข่ายจากลูกข่ายไปยังแม่ข่าย เพื่อป้องกันการรั่วไหลของข้อมูลจากสายเคเบิล

ถ้าหากแม่ข่ายของ Netware มีการทำงานเป็น Non-dedicated คือใช้เป็นที่แม่ข่ายและลูกข่ายบนเครื่องเดียวกัน ผู้ใช้ที่ทำงานอยู่ที่เครื่องแม่ข่ายจะต้องมีการ Login ก่อนเข้าไปทำงานในระบบเสมอ ซึ่งหมายความว่าไม่มีใครสามารถเข้าไปเรียกดูข้อมูลในระบบโดยไม่ผ่านการ Login

การ Login และระบบรักษาความปลอดภัยของ Netware ไม่อนุญาตให้เข้าไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 การเชื่อมโยงระบบเครือข่ายในลักษณะ internal router

ส่วนประกอบของระบบเครือข่าย NetWare

NetWare เป็นการเชื่อมต่อคอมพิวเตอร์ส่วนบุคคลและอุปกรณ์อื่นๆ เช่น เครื่องพิมพ์และดิสก์ไดรฟ์เข้าด้วยกัน ทำให้ผู้ใช้สามารถใช้อุปกรณ์และไฟล์ร่วมกันส่งข่าวสารจากเครื่องหนึ่งไปยังอีกเครื่องหนึ่ง และป้องกันไฟล์ที่เป็นความลับโดยกำหนดระดับของผู้ใช้ได้

ระบบเครือข่าย NetWare ประกอบด้วย ศูนย์บริการข้อมูล(file server) สถานีผู้ใช้(work station) อุปกรณ์ประกอบอื่นๆ ที่ต่อเชื่อมกับศูนย์บริการข้อมูล

ศูนย์บริการข้อมูลหลายตัวอาจจะเชื่อมต่อกันบนระบบเครือข่ายกันเป็นระบบเครือข่าย แบบ multiserver และหลายระบบเครือข่ายก็อาจเชื่อมต่อกันโดยผ่านฮาร์ดแวร์ และ ซอฟต์แวร์ bridge เพื่อทำเป็นระบบเครือข่ายร่วม (internetwork) ได้

ศูนย์บริการข้อมูล เป็นหัวใจของระบบเครือข่าย NetWare ทำหน้าที่จัดการเกี่ยวกับการใช้ไฟล์ร่วมกัน ระบบรักษาความปลอดภัยของข้อมูล การทำงานติดต่อกันระหว่างสถานี การใช้ดิสก์ร่วมกัน ผู้ใช้ระบบเครือข่ายอาจตรวจสอบและควบคุมการทำงานของศูนย์บริการ ข้อมูลแต่ละตัวได้จากจอภาพของศูนย์บริการข้อมูล ผู้ใช้สามารถเริ่มและหยุดศูนย์บริการข้อมูล กำหนดเวลาการทำงาน ควบคุมการทำงานอื่นๆ ได้

5 ตัวและเครื่องหมายเหล่านี้จะเป็นเครื่องหมายของระบบ ซึ่งสถานีผู้ใช้ใดๆ สามารถติด ต่อใช้งาน ได้โดย เรียกผ่านศูนย์บริการข้อมูล

ดิสก์ของระบบเครือข่าย ศูนย์บริการข้อมูลแต่ละตัวอาจมีฮาร์ดดิสก์ตั้งแต่ 1 ตัวขึ้นไป ฮาร์ดดิสก์ที่ถูกควบคุมโดย server เรียกว่า ดิสก์ของระบบเครือข่าย (network disks) ดิสก์ของระบบเครือข่ายจะแบ่งเป็น Volume ได้เรกทอรีย่อย และไฟล์

- Volume ศูนย์บริการข้อมูลแต่ละตัวจะต้องมีที่เก็บข้อมูลอย่างน้อย 1 Volume คือ sys:VOLUME ซึ่งจะถูกสร้างขึ้นเมื่อทำการติดตั้งศูนย์บริการ นอกเหนือจากนี้ผู้ใช้อาจสร้าง Volume อื่นๆ หรือแบ่งเนื้อที่บนดิสก์เป็นหลาย Volume ก็ได้

- ไดเรกทอรี แต่ละ Volume จะแบ่งหน่วยทางลอจิกที่เรียกว่าไดเรกทอรี (directory) ในไดเรกทอรีหนึ่งๆ อาจประกอบด้วย ไดเรกทอรีย่อยได้อีก และในไดเรกทอรีย่อยอาจมีแยกย่อย ออกไปได้อีกเรื่อยๆ ดังนั้นผู้ติดตั้งระบบจะเป็นผู้กำหนดจำนวนไดเรกทอรีสูงสุดที่มีในแต่ละ Volume ในระหว่างการติดตั้งระบบและหลังจากทำการติดตั้งระบบแล้ว ผู้ควบคุมระบบจะเป็นผู้ที่กำหนดสิทธิ ในการใช้ไดเรกทอรีต่างๆ บนศูนย์บริการข้อมูลให้กับผู้ใช้แต่ละคน ผู้ใช้บางคนจะสร้างไดเรกทอรี หรือไฟล์เพิ่มขึ้นอีกก็ได้ แต่บางคนใช้งานได้อย่างเดียวไม่สามารถที่สร้างเพิ่มเติมได้ เป็นต้น

- ไฟล์ จะเป็นหน่วยย่อยของ ไดเรกทอรี

สถานีผู้ใช้ ผู้ใช้สามารถติดต่อกับศูนย์บริการข้อมูลโดยผ่านสถานีผู้ใช้ซึ่งก็ หมายถึง คอมพิวเตอร์โดยทั่วไป ซึ่งจะประกอบไปด้วย จอภาพ แป้นพิมพ์ ดิสก์ของสถานีผู้ใช้ (local disk) หน่วยประมวล (cpu) สถานีผู้ใช้แต่ละตัวจะติดต่อกับระบบปฏิบัติการของศูนย์บริการ โดยผ่าน network shell ซึ่งจะทำให้ผู้ใช้สามารถจะใช้งานศูนย์บริการข้อมูลได้

ดิสก์ประจำสถานีผู้ใช้ จะหมายถึง หน่วยบันทึกข้อมูลของผู้ใช้ที่อยู่ที่สถานีผู้ใช้ ซึ่งไม่ใช่โครงสร้างของระบบเครือข่าย ซึ่งในกรณีของสถานีผู้ใช้ด้วย Dos แล้ว Dos จะกำหนดอักษรหมายเลขไดรฟ์ ให้ใช้ไปยัง ไดรฟ์ของผู้ใช้ซึ่งโดยปกติจะเป็น เช่น ไดรฟ์ A ไดรฟ์ B เป็นต้น

ไดรฟ์ของระบบเครือข่าย ในระบบเครือข่ายสามารถกำหนดอักษรหมายเลขไดรฟ์ให้ใช้ไปยัง ไดเรกทอรีของระบบเครือข่ายได้ ซึ่งไดรฟ์ดังกล่าวนี้จะเรียกว่า ไดรฟ์ของระบบเครือข่าย (network drive) เช่น ถ้าไดรฟ์ F และ G ใช้ไปยังไดเรกทอรี ของระบบเครือข่าย แล้วจะ เรียกไดรฟ์ F ไดรฟ์ G ว่าเป็นไดรฟ์ของระบบเครือข่าย ดังนี้ เป็นต้น โดยปกติระบบเครือข่าย

การจัดระบบไดเรกทอรี

เมื่อจัดตั้งระบบเครือข่าย เราจะตั้งชื่อให้แก่ ศูนย์บริการข้อมูลและสร้างโวลูมในฮาร์ดดิสก์ของศูนย์บริการข้อมูลจากนั้นผู้ใช้จึงจะสามารถสร้างโครงสร้างของไดเรกทอรีได้

การกำหนดชื่อไดเรกทอรี ในชื่อไดเรกทอรีหนึ่งๆ อาจแบ่งได้เป็นหลายส่วนคือ ศูนย์บริการข้อมูล Volume ไดเรกทอรีและไดเรกทอรีย่อย การใช้งานไฟล์ที่เก็บอยู่ในไดเรกทอรีใดๆ จะต้องกำหนดชื่อและลำดับของส่วนต่างๆ เหล่านี้ให้ถูกต้อง มีวิธีกำหนดชื่อไฟล์ใน

ไดเรกทอรีได้ 2 แบบ

1. แบบเต็ม (full directory names) เป็นการกำหนดแต่ละส่วนที่เป็นไปได้ในชื่อไดเรกทอรี

รูปแบบ

servername/volume:directory/directory/.../filename

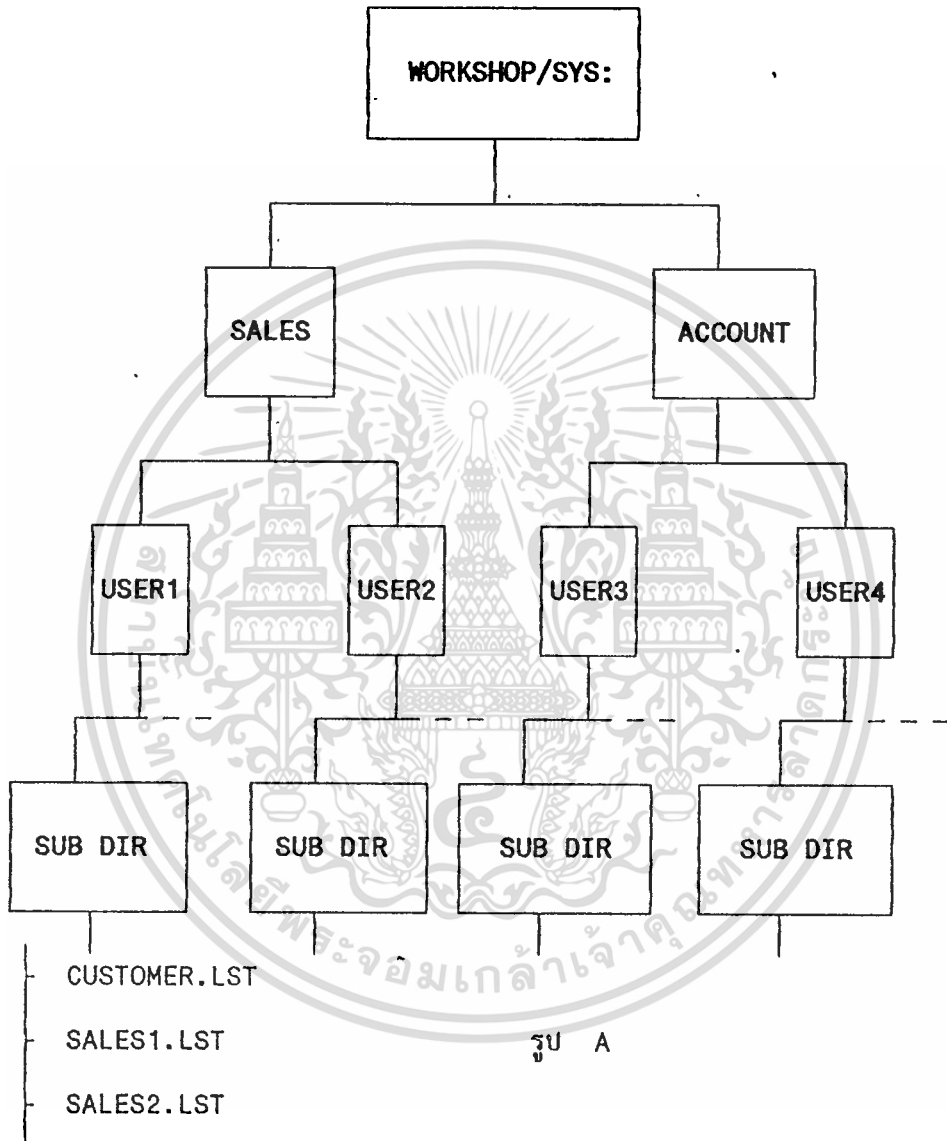
2. แบบสัมพันธ์ (relative directory names) การกำหนดแบบนี้ จะสะดวกและสั้นกะทัดรัดกว่าการใช้แบบเต็ม

รูปแบบ

filename

ถ้าในระบบเครือข่าย มีศูนย์บริการข้อมูลเพียงตัวเดียวแล้ว โดเมนของระบบเครือข่ายจะกำหนดให้แก่ไดเรกทอรีบนศูนย์บริการตัวเดียว ในกรณีนี้ไม่จำเป็นต้องระบุชื่อของ ศูนย์บริการเราอาจกำหนดชื่อแบบสัมพันธ์ได้

ตัวอย่าง สมมติว่าเรามีโครงสร้างไดเรกทอรีดังรูป A



การกำหนดชื่อแบบเต็มของไฟล์ SALES1.LST ของ USER1 เป็นดังนี้

WORKSHOP/SYS:SALES/USER1/SUBDIR/SALES1.LST

ถ้าใคร่ของระบบเครือข่ายที่ใช้อยู่กำหนดเป็น WORKSHOP/SYS:SALES/USER
/REPORT แล้วเราสามารถกำหนดชื่อไฟล์ได้ดังนี้

เอกสารนี้เป็นเอกสาร SALES1.LST สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกำหนดไดรฟ์ของระบบเครือข่าย

เมื่อได้ทำความเข้าใจเกี่ยวกับ การกำหนดชื่อไดเรกทอรีแล้ว ขั้นตอนต่อไปจะทำความเข้าใจเกี่ยวกับการกำหนดไดรฟ์ให้ระบบเครือข่าย หรือ map ไดรฟ์ของระบบเครือข่ายให้แก่ไดเรกทอรี การกำหนดทำให้เราทำงานกับไดเรกทอรีใดๆ ได้ โดยไม่ต้องบอกชื่อเต็ม ของไดเรกทอรีนั้น โปรแกรมที่ใช้ในการกำหนดไดรฟ์ให้แก่ไดเรกทอรีมีชื่อว่า MAP ตัวอย่างการกำหนดไดรฟ์ G ให้แก่ไดเรกทอรี WORKSHOP/SYS:SALES/USER1/REPORT กำหนดได้ดังนี้

```
MAP G:= WORKSHOP/SYS:SALES/USER1/SUBDIR <enter>
```

หลังจากไดรฟ์ G กำหนดให้เรียบร้อยแล้วไดเรกทอรี WORKSHOP/SYS:SALES/USER1/REPORT แล้วเราสามารถกำหนดไดรฟ์ใช้งานปัจจุบัน (default drive) ให้เป็นไดรฟ์ G ได้โดย

```
G: <enter>
```

เมื่อเป็นไดรฟ์ G แล้วไดเรกทอรีปัจจุบันจะกลายเป็น WORKSHOP/SYS:SALES/USER1/REPORT และในการใช้งานไฟล์ในไดเรกทอรีนี้ ไม่จำเป็นต้องบอกแบบชื่อเต็มให้ระบุชื่อไฟล์ได้เลย

เรากำหนดไดรฟ์ใช้งานได้มากที่สุด 26 ไดรฟ์ ไปยังไดเรกทอรีของระบบเครือข่ายแตกต่างกัน สำหรับในระบบเครือข่ายที่มีศูนย์บริการหลายแห่งรวมทั้งในระบบเครือข่ายร่วม (internet-work) นั้นอาจกำหนดไดรฟ์ไปยังไดเรกทอรีของศูนย์บริการข้อมูลได้มากถึง 8 แห่งด้วยกัน ปกติแล้วการกำหนดไดรฟ์มักทำกับไดเรกทอรีที่ใช้บ่อยๆ เมื่อต้องการเปลี่ยนไปยังไดเรกทอรีนี้เพียงแต่ป้อนตัวอักษรที่กำหนดมานี้ก็สามารถอ้างถึงไดเรกทอรีนั้นได้ทันที

ไดรฟ์สำหรับการค้นหาข้อมูล ปกติเมื่อมีการถามเพื่อขอใช้งานไฟล์บนระบบเครือข่าย ระบบปฏิบัติการของระบบเครือข่ายจะมองหาไฟล์ในไดเรกทอรีปัจจุบันก่อนถ้าหาไม่พบก็จะแสดงข้อความออกมาว่าหาไฟล์ไม่พบ เรากำหนดให้ค้นหาไฟล์โปรแกรมในไดเรกทอรีอื่นได้โดยการกำหนดไดรฟ์สำหรับการค้นหาไดรฟ์สำหรับการค้นหานี้จะอนุญาตให้ระบบปฏิบัติการค้นหาไฟล์โปรแกรมในไดเรกทอรีอื่นนอกเหนือจากไดเรกทอรีปัจจุบันได้ ถ้ามีการกำหนดไดรฟ์สำหรับการค้นหา และระบบหาไฟล์ไม่พบในไดเรกทอรีปัจจุบันแล้ว ระบบจะมองหาไฟล์โปรแกรมในไดเรกทอรีที่กำหนดให้เป็นไดรฟ์สำหรับการค้นหาทั้งหมดจะกระทั่งพบหรือได้ค้นหาครบทุกไดเรกทอรีที่กำหนดเป็นไดรฟ์สำหรับการค้นหา ในการกำหนดไดรฟ์สำหรับการค้นหาเราสามารถกำหนดได้สูงสุด 16 ไดรฟ์

สคริปต์คำสั่งตอน LOGIN

จากที่ผ่านมามองเห็นแล้วว่า การใช้ไครฟ์ที่กำหนดและไครฟ์สำหรับการค้นหาช่วยประหยัดเวลาลงได้มาก แต่อย่างไรก็ดี การกำหนดเหล่านี้จะไม่คงอยู่อย่างถาวรเมื่อเรา LOGOUT ออกจากระบบ ไครฟ์ที่กำหนดการใช้งานเหล่านั้นก็จะหายไป

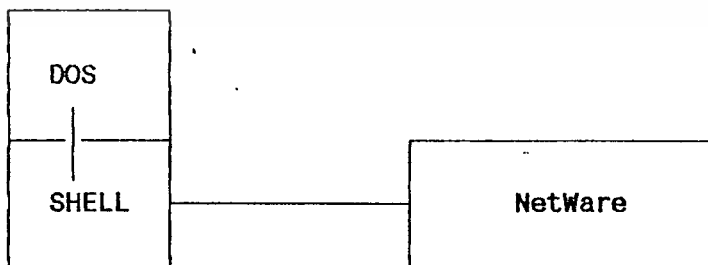
เราสามารถป้องกันการกำหนดไครฟ์ใช้งานแต่ละครั้งที่ LOGIN เข้ามาในระบบได้แต่ก็จะยุ่งยากและเสียเวลามาก มีอีกวิธีหนึ่งที่จะสามารถนำเราคำสั่งการกำหนดไครฟ์ใช้งานมาใช้งานได้เมื่อ LOGIN เข้าระบบ นั่นคือการใช้ สคริปต์คำสั่งตอน LOGIN

การป้อนข้อความไปเก็บในสคริปต์คำสั่งทำโดยเรียกผ่านเมนู user information ในโปรแกรม SYSCON ซึ่งจะช่วยให้ระบบกำหนดไครฟ์ใช้งานและไครฟ์สำหรับการค้นหาที่ต้องการให้โดยอัตโนมัติเมื่อผู้ใช้งานมีการ LOGIN เข้าระบบ

คำสั่งในสคริปต์คำสั่งนั้นมีคำสั่งที่ทำให้ผู้ใช้ติดต่อกับศูนย์บริการข้อมูลตัวอื่นได้ แต่ถ้าเรา LOGIN เข้าศูนย์บริการข้อมูลตัวแรกแล้ว LOGIN เข้าศูนย์บริการอีกแห่งหนึ่ง สคริปต์คำสั่งของผู้ใช้ในศูนย์บริการตัวที่สองจะไม่ถูกเรียกขึ้นมาทำงานด้วย

การทำงานของ NetWare

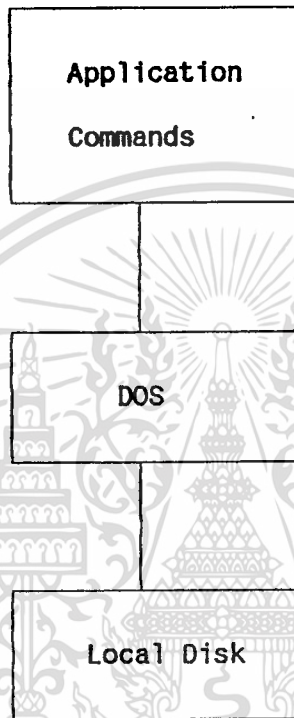
ระบบปฏิบัติการ NetWare สนับสนุนการใช้งานเครื่อง ไมโครคอมพิวเตอร์ที่ต่างกันหรือมีระบบปฏิบัติการที่ต่างกันโดยไม่ต้องมีการแบ่งเนื้อที่บนดิสก์ ไฟล์ทั้งหมด และสถานีผู้ใช้ทุกสถานีสามารถใช้งานใด เรกทอรีร่วมกันได้



รูปที่ 2.23 Shell ของ Netware ทำหน้าที่ติดต่อระหว่างดอสของสถานีผู้ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

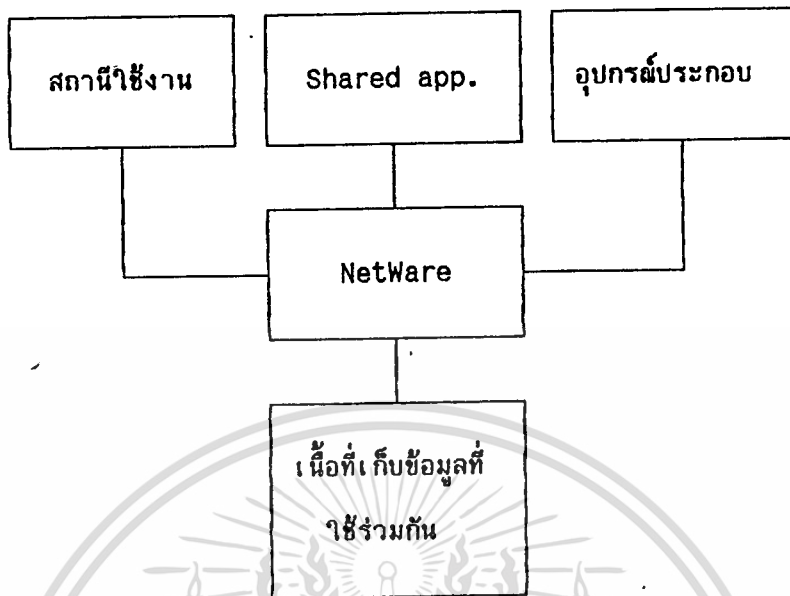
NetWare จะอยู่ที่ศูนย์บริการข้อมูลและ Dos จะอยู่ที่สถานีผู้ใช้ NetWare Shell อยู่ที่สถานีของผู้ใช้เพื่อให้ Dos ติดต่อกับ NetWare ได้



รูปที่ 2.24 การทำงานของ DOS (สภาวะแวดล้อมแบบมีผู้ใช้คนเดียว)

Dos, NetWare และ Shell จะทำงานร่วมกันเพื่อช่วยในการใช้งานข้อมูลและฮาร์ดแวร์ร่วมกัน Dos ออกแบบให้ประมวลผลโปรแกรม ที่สถานีผู้ใช้ในสภาพแวดล้อมแบบผู้ใช้คนเดียว (single-user environment)

NetWare ถูกออกแบบให้ประมวลผลในสภาพแวดล้อมที่มีผู้ใช้หลายคน (multi-user environment) ซึ่งจะช่วยในการใช้งานอุปกรณ์ต่างๆ ร่วมกัน ใช้งานโปรแกรมร่วมกัน ควบคุมไดเรกทอรีที่ซับซ้อนและ file-allocation table (FAT) ที่ศูนย์บริการข้อมูล



รูปที่ 2.25 การทำงานของ Netware (สภาวะแวดล้อมแบบมีผู้ใช้หลายคน)

การรักษาความปลอดภัยและการจัดการไฟล์ที่ Dos ไม่ได้จัดการให้จะถูกจัดการโดย NetWare ที่ศูนย์บริการข้อมูลโดย NetWare กำหนดบุคคลที่มีคุณสมบัติดังนี้

- ผู้ที่สามารถใช้งานศูนย์บริการได้
- ผู้ที่สามารถใช้งานไฟล์ได้
- ผู้ที่สามารถจะใช้งานไฟล์ร่วมกันได้

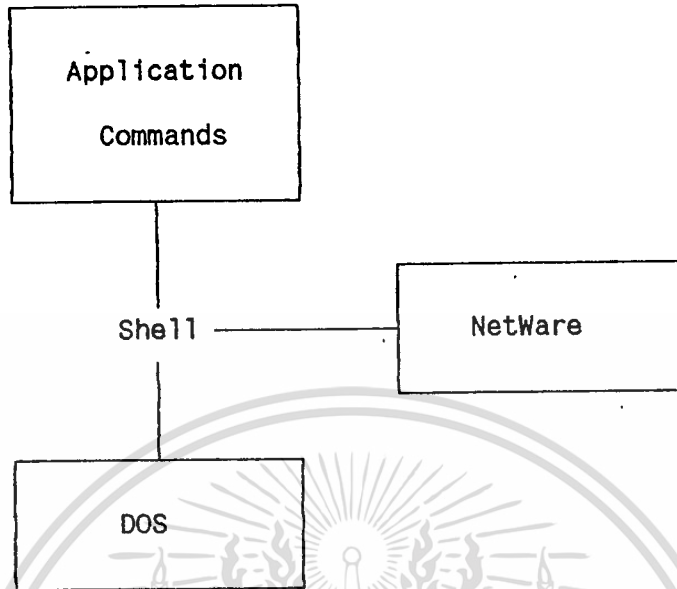
NetWare นั้นไม่ใช่ Dos ในสภาพแวดล้อมของระบบเครือข่ายแต่ Dos จะถูกใช้ ที่สถานีผู้ใช้เท่านั้น เพื่อทำงานบนไครท์ที่เป็นของตัวเองรวมทั้ง โปรแกรมของผู้ใช้

NetWare ควบคุมการสร้าง และจัดการข้อมูลบนศูนย์บริการ ในขณะที่ NetWare Shell จะช่วย Dos และ NetWare สามารถติดต่อกันได้

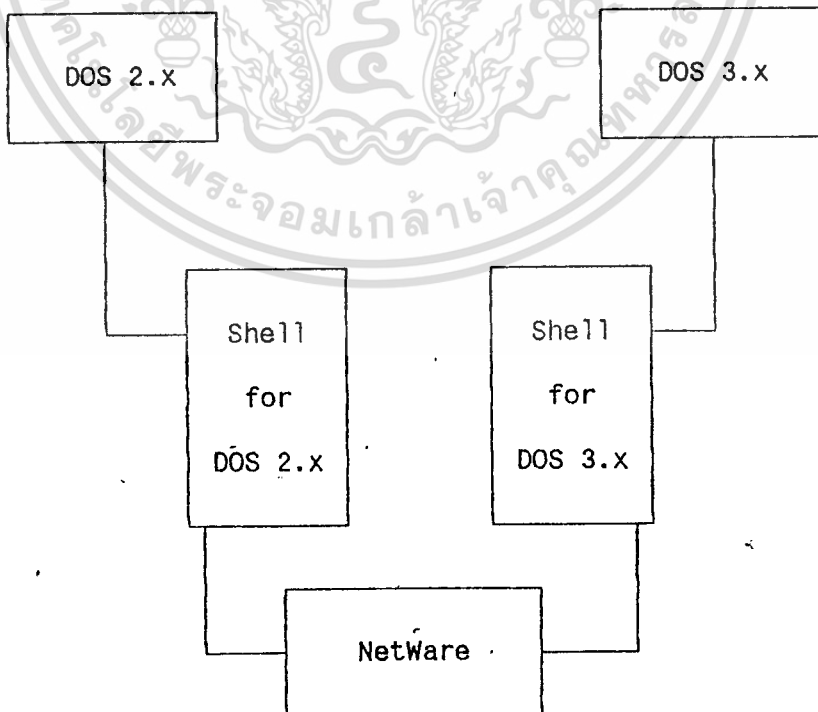
NetWare Shell ซึ่งมีขนาดประมาณ 40 กิโลไบต์ จะอยู่ที่สถานีของผู้ใช้และดักการเรียกอินเตอร์พรีต 21h (การขอข้อมูลหรือคำสั่งของ NetWare ที่อยู่บนศูนย์บริการข้อมูล) .

Dos จะเปรียบได้กับผู้ที่พูดและเข้าใจเฉพาะภาษาอังกฤษ NetWare เป็นผู้ฟังและเข้าใจเฉพาะภาษาไทย Shell จะเป็นล่ามระหว่าง Dos กับ NetWare เป็นผลให้ทั้ง 2 สามารถติดต่อและเข้าใจกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.26 การทำงานของ Shell ภายใต้ Netware



รูปที่ 2.27 การสนับสนุนการใช้ DOS ภายใต้ Netware

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การสงวนเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดก็ตาม ผู้ใช้ต้องรับผิดชอบต่อการใช้งาน และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

shell จะทำให้ NetWare ต่างไปจากระบบปฏิบัติการของระบบเครือข่ายอื่น เพราะว่าไม่จำกัดว่าสถานะของผู้ใช้ต้องใช่ Dos รุ่นไหน NetWare จะสนับสนุน Dos ตั้งแต่ รุ่น 2.0 ขึ้นไป หมายความว่า สถานะหนึ่งอาจใช้ Dos 2.0 และสถานะอื่นในระบบเครือข่ายเดียวกันอาจใช้ Dos 3.1 หรือ 3.2 ก็ได้

การรับรองมาตรฐานต่างๆ

Novell มีนโยบายเด่นชัดที่จะรองรับมาตรฐานต่างๆ เพราะการรองรับมาตรฐานของ Netware จะสามารถเอื้ออำนวยให้ผู้ใช้งานสามารถค่อยๆ เปลี่ยนไปใช้เทคโนโลยีใหม่ๆ โดยไม่จำเป็นต้องทิ้งของเก่า พร้อมทั้งยังช่วยให้ระบบคอมพิวเตอร์หลากหลายชนิดสามารถทำงานร่วมกันได้ ซึ่งเป็นการเพิ่มประสิทธิภาพ และคุณค่าของระบบเครือข่าย มาตรฐานที่รองรับโดย Netware มีทั้งมาตรฐานทางด้านซอฟต์แวร์และฮาร์ดแวร์ เช่น

- Interprocess Communication Protocol (IPC) เช่น Sequenced Package Exchange (SPX) ของ Novell, NetBIOS และ APPC ของ IBM, NAME Pipes ของ Microsoft, Transport Level Interface (TLI) ของ AT&T และ BSD Sockets
- Transport Protocol เช่น SPX/IPX ของ Novell, APPLE TALK ของ APPLE TCP/IP และ OSI
- เน็ตเวิร์คฮาร์ดแวร์ เช่น IEEE 802.3, Ethernet, IEEE 802.5 Token-Ring และ ArcNet

2.3 วิธีการเชื่อมต่อระบบหรือโทโพโลยี (Topology)

วิธีการเชื่อมต่อระบบหรือโทโพโลยี เป็นรูปแบบการต่อของระบบ LAN หรือ รูปร่างของ เน็ตเวิร์ค โดยโครงสร้างแบบที่สำคัญ เช่น แบบดาว (star), แบบบัสเชิงเส้น (linear bus) และแบบวงแหวน (ring) เรากำหนดโทโพโลยี เหล่านี้ได้ 2 ทาง คือ

1. โทโพโลยีเชิงตรรก (Logical Topology) เป็นวิธีที่สถานะของระบบเครือข่ายติดต่อกับสื่อและข้อมูล
2. โทโพโลยีทางกายภาพ (Physical Topology) เป็นวิธีทางกายภาพที่ระบบเครือข่ายถูกเชื่อมต่อ.

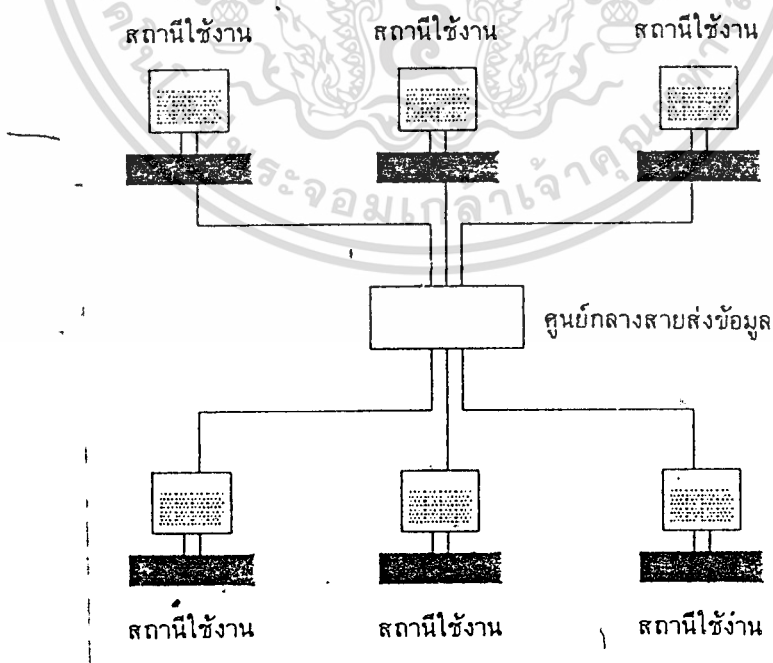
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.1 โทโพโลยีแบบดาว (star)

จะต่อสายส่งข้อมูลในรูปแบบดาว ดังรูป นั่นคือ มีสายต่อออกมาจากศูนย์กลางซึ่งปกติจะเป็น ศูนย์บริการข้อมูล (File server) แต่ละสถานีจะมีสายต่อเข้ากับ ศูนย์บริการ ไม่มีการใช้สายส่งข้อมูลร่วมกัน ดังนั้นถ้าสายส่งข้อมูลของสถานีใด เสียหาย ก็จะมีผลเฉพาะสถานีนั้นเท่านั้น ไม่มีผลต่อสถานีอื่น

- ข้อดี
- โทโพโลยีแบบดาวมีจุดศูนย์กลางอยู่ที่เดียว ซึ่งง่ายในการให้บริการหรือติดตั้ง ระบบ
 - อุปกรณ์ 1 ตัวต่อกับสายส่งข้อมูล 1 เส้น ทำให้การเสียหายของอุปกรณ์ในระบบไม่กระทบกระเทือนต่อการทำงานของจุดอื่นๆ ในระบบ
 - ใช้โปรโตคอล (protocol) แบบง่ายๆ การเชื่อมต่อในระบบที่ใช้โทโพโลยีแบบดาว จะเกี่ยวข้องกันระหว่างศูนย์กลางและอุปกรณ์ที่อีกจุดหนึ่งเท่านั้น ทำให้ควบคุมการส่งข้อมูลทำได้ง่าย

- ข้อเสีย
- เนื่องจากแต่ละจุดต่อโดยตรงกับศูนย์กลางจึงต้องใช้สายส่งข้อมูลจำนวนมากในการติดตั้งระบบ ยิ่งระบบกว้างเท่าไรก็ต้องเสียค่าใช้จ่ายในการติดตั้งและบำรุงรักษาเท่านั้น



รูปที่ 2.28 โทโพโลยีแบบดาว

- การขยายระบบทำได้ลำบาก คือ ในการเพิ่มจุดใหม่เข้าไปในระบบจำต้องเดินสายส่งข้อมูลจากศูนย์กลางออกมา

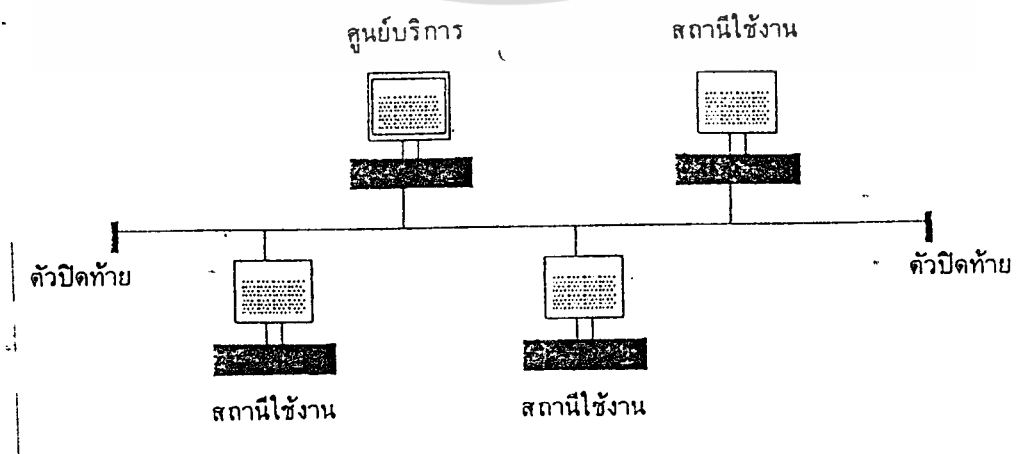
2.3.2 โทโพโลยีแบบบัส (Bus Topology)

มีสายส่งข้อมูลอันเดียวเรียกว่า บัส หรือ ทังก์ (trunk) ซึ่งทุกสถานีจะมีสายส่งต่อกับบัสนี้และที่ปลายทั้งสองของบัสปิดด้วยตัวเทอร์มิเนเตอร์ ดังรูป โดยปกติแต่ละสถานีจะต่อด้วยสายส่งข้อมูลหลักด้วย stubs ในการใช้งานจริง สายส่งข้อมูลหลักมักจะอยู่บนเพดานหรือติดอยู่กับผนัง และ stubs จะเป็นตัวต่อจากสายข้อมูลหลักเข้าสู่สถานีของผู้ใช้ ตัว stubs มีความยาวได้ค่อนข้างจำกัด เช่น ของ Ethernet จะมีความยาวสูงสุดได้แค่ 2.5 เมตร เป็นต้น

โทโพโลยีแบบบัสมีข้อดีที่ การเดินสายง่ายและประหยัด แต่ถ้ามีการต่อสายข้อมูลผิด หรือ ด้านใดด้านหนึ่งของบัสไม่มีตัวปิด หรือ ตัว เทอร์มิเนเตอร์ปิดอยู่จะทำให้ระบบไม่สามารถทำงานได้ ข้อเสียอีกอย่างหนึ่ง คือ ในการที่ระบบ ทำงานผิดปกติ จะตรวจสอบหาข้อผิดพลาดได้ยาก โดยเฉพาะถ้าเป็นระบบขนาดใหญ่

มีรูปแบบหนึ่งของบัสที่พบอยู่ในปัจจุบัน คือ การเชื่อมต่อของสถานี ในรูปแบบดาว แต่มีการทำงานทางไฟฟ้าแบบบัส เช่น แบบ Star Lan ของบริษัท AT&T

- ข้อดี
- ใช้สายส่งข้อมูลน้อย และมีรูปแบบข้อมูลที่ง่าย คือ มีการใช้สายส่งข้อมูลร่วมกันทำให้ใช้สายส่งข้อมูลได้อย่างประหยัด ลดค่าใช้จ่ายในการติดตั้งและบำรุงรักษา
 - มีรูปแบบง่ายและมีความน่าเชื่อถือ เพราะใช้สายส่งเพียงเส้นเดียวเป็นเส้นหลัก

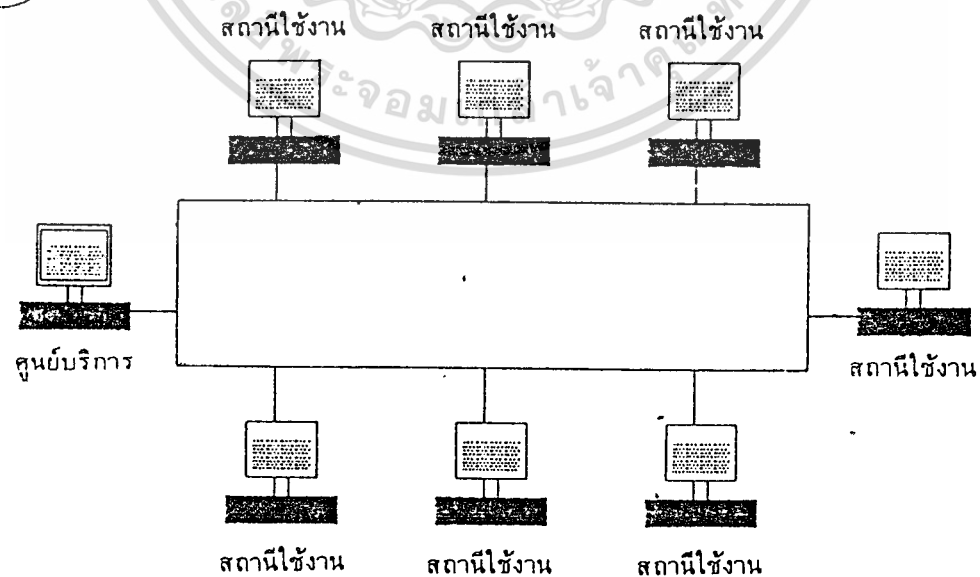


รูปที่ 2.29 โทโพโลยีแบบบัส

- ข้อเสีย
- การหาข้อผิดพลาดของระบบทำได้ยาก เนื่องจาก การควบคุมระบบไม่มีศูนย์กลางอยู่ที่จุดใดจุดหนึ่ง โดยเฉพาะดังนั้นการตรวจสอบข้อผิดพลาดต้องทำทุกๆจุดในระบบ
 - ในกรณีที่เกิดการเสียหายในสายส่งข้อมูลสายใดสายหนึ่ง จะทำให้ทั้งระบบไม่สามารถทำงานได้
 - เมื่อมีการขยายระบบแบบบัสโดยใช้ตัวทวนซ้ำสัญญาณ จะต้องมีการจัดโครงสร้างระบบใหม่ ทำให้เกิดความเสียหาย และเสียค่าใช้จ่ายสูง
 - เนื่องจากในระบบไม่มีศูนย์กลางในการควบคุมระบบ ดังนั้นการตัดสินใจว่าสถานีใดจะใช้สายส่งข้อมูลจะเป็นหน้าที่ของแต่ละสถานี ทำให้เกิดค่าใช้จ่ายเพิ่มขึ้นทั้งฮาร์ดแวร์ และ ซอฟต์แวร์

แต่ละสถานีจะต่อกันเป็นลูป (loop) ดังนั้น สถานีแรกจะต่อกับสถานีสุดท้าย ทำให้รูปแบบของระบบเป็นวงแหวน ข้อมูลที่ส่งในระบบจึงต้องผ่านทุกสถานี

- ข้อดี**
- ใช้สายส่งข้อมูลน้อย ทำให้เพิ่มความน่าเชื่อถือของระบบได้มากขึ้น
 - ถ้าใช้กับสายส่งข้อมูลเส้นใยนำแสง สามารถส่งข้อมูลด้วยความเร็วสูง



ข้อเสีย - ถ้าสถานีใดสถานีหนึ่งเสีย จะทำให้ทั้งระบบไม่สามารถทำงานได้

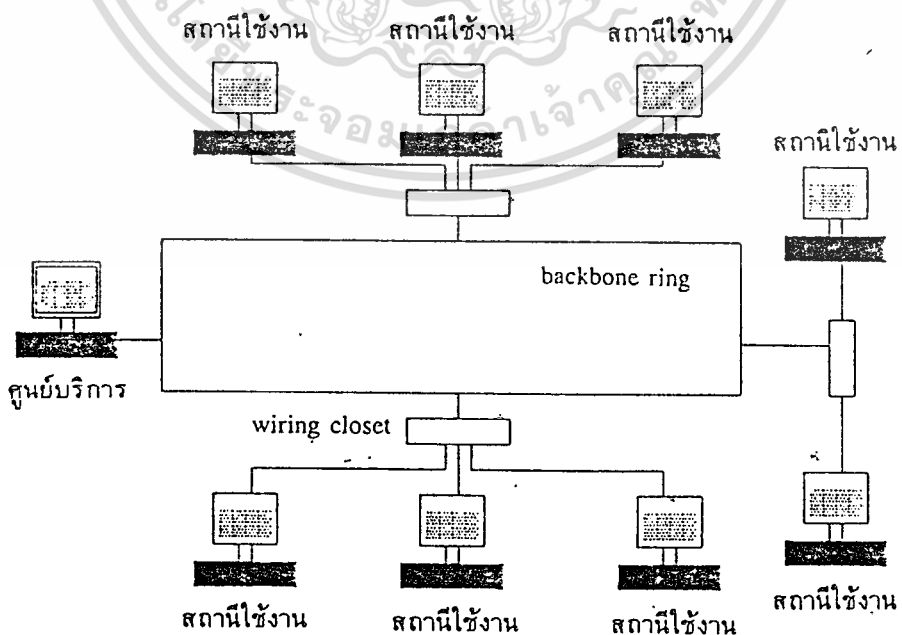
- ยากในการตรวจหาข้อผิดพลาด

- ในการเพิ่มสถานีใหม่ เข้าไปในระบบ จะต้องจัดระบบเชื่อมโยงข้อมูลใหม่

2.3.4 โทโพโลยีแบบวงแหวน-ดาว (Star-Ring Topology)

ระบบ LAN ที่มีระบบโทโพโลยีแบบวงแหวน-ดาว ดังรูป จะต่อกันทางกายภาพแบบดาวที่ต่อออกจากวงแหวนของฮับ (Hub) ดังนั้น แต่ละสถานีจะถูกมองเหมือนว่าต่อเป็นแบบวงแหวน โดยมีฮับเป็นตัวช่วยในการเชื่อมต่อกับสถานีที่ไกลที่สุด ตัวอย่างของโทโพโลยีแบบนี้ ก็คือ ระบบเครือข่ายแบบ Token-Ring ของ IBM

- ข้อดี
- ง่ายในการตรวจสอบข้อผิดพลาดเนื่องจากโทโพโลยีแบบนี้มีไว้ Wiring Closet ซึ่งเมื่อเกิดข้อผิดพลาดขึ้น ก็จะตรวจสอบได้ว่าเกิดกับ Wiring Closet ไດในวงแหวน ซึ่งก็จะแยก Wiring Closet นั้น มาทำการแก้ไข ซึ่งก็จะไม่กระทบกระเทือนต่อการทำงานของระบบเครือข่าย, ง่ายต่อการขยายระบบ
 - การเดินสายส่งข้อมูล Concentration Point ในระบบที่ใช้โทโพโลยีแบบวง



แหวน-ดาว จะเชื่อมต่อกันโดยผ่านสายส่งข้อมูลเพียงสายเดียว ทำให้สะดวกต่อการติดตั้ง

ข้อเสีย - Concentration Point จะต้องมีความสามารถในการประมวลผลที่ดีพอ ซึ่งจะอยู่กับการนำระบบไปใช้งานด้วย จึงจะทำให้ระบบเครือข่ายแบบนี้มีประสิทธิภาพในการทำงานสูง

- การเดินสายอาจจะต้องใช้สายส่งข้อมูลซ้ำซ้อน เพื่อใช้เป็นวงแหวนสำรอง ถ้าต้องการความเชื่อถือของระบบเพิ่มขึ้น

2.4 โพรโทคอล (Protocol)

ตามกำหนดโดย ISO (International Organization for Standardization) ได้แบ่งระดับการสื่อสารข้อมูลออกเป็น 7 ระดับ สำหรับใช้เชื่อมต่อกับระบบเปิด (Open System Interconnection : OSI) ที่สามารถใช้กับ Protocol ที่มีอยู่ต่างๆ ได้ โดยรูปแบบของ model OSI แบบ 7 ชั้น เป็นดังนี้

Application layer	(7)
Presentation layer	(6)
Session layer	(5)
Transport layer	(4)
Network layer	(3)
Data link layer	(2)
Physical layer	(1)

The physical layer

ชั้นที่ 1 จะประกอบด้วยตัวที่จ้องทำหน้าที่ active, maintain, deactivate สำหรับการติดต่อกันทางกายภาพ (physical) มันกำหนดว่าใช้ voltages เท่าไหร่ และ ระดับสัญญาณ

เป็นอย่างไร เช่น เมื่อมันถูกใช้งาน มันไม่ต้องกล่าวถึงตัวกลางที่ใช้ในการสื่อสารหรือ การ modulation ว่าเป็นอย่างไรบ้าง

ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The data link layer

ชั้นที่ 2 จะครอบคลุม การกำหนดสัญญาเวลาให้พร้อมกัน และ ควบคุมความผิดพลาด สำหรับข้อมูลข่าวสารต่างๆ ที่ถูกส่งมาจาก physical link โดยไม่คำนึงถึงขอบเขต

The network layer

ชั้นที่ 3 จะเป็นตัวกระทำเส้นทางของการสื่อสาร ตลอดจนแหล่งที่มาของการสื่อสารที่แตกต่างกัน เช่น มาจากช่องที่ต่างกัน (different channels) มันจะทำหน้าที่เป็นตัวควบคุม network โดยทำการตัดสินใจว่า ทางไหนที่ควรจะให้ข้อมูลผ่านไปโดยครอบคลุมตั้งแต่มาจาก physical network จนถึงการประมวลผลทางด้าน application(user) ข้อมูลซึ่งถูกกำหนดอยู่ระหว่าง network หรือจาก node หนึ่งไปยังอีก node หนึ่งภายใน network นั้น ต้องการใช้เพียงแต่หน้าที่ของ layer 1 ถึง layer 3 เท่านั้น ดังแสดงในรูปที่ 11-22 network นั้นอาจจะกระทำเสมือนเป็น relay link โดยข้อความต่างๆจะผ่านจาก layer 1 ขึ้นไป 3 และลงมาอีกครั้งหนึ่ง

The transport layer

ชั้นที่ 4 จะรับผิดชอบการรวมกลุ่มของข้อความที่ไม่เกี่ยวข้องกันหรือเป็นอิสระต่อกันใน single connection เมื่อ user หลายคนต้องการจะส่งข้อมูลหรือข้อความในทิศทางเดียวกัน มันจะทำการโดยแตกกลุ่มของข้อมูลออกเป็นกลุ่มย่อยๆ หรือกลุ่มเล็กๆ เพื่อที่ว่าจะสามารถส่งไปโดย network layer อย่างมีประสิทธิภาพ และมันจะรับประกันว่าข้อมูลจากปลายข้างหนึ่งจะสามารถส่งไปยังปลายอีกด้านหนึ่งใน network ได้

The session layer

ชั้นที่ 5 จะจัดการและจัดเวลาของการสนทนากันระหว่าง 2 applications หรือ user ที่แตกต่างกัน ดังตัวอย่าง เช่น ชุดของข้อมูลจะถูกจัด และ กำหนดขนาดของข้อมูลไว้อย่างเรียบร้อยแล้ว ดังนั้นที่ปลายทางทั้งข้อมูลจะถูกส่งไปอย่างเต็มส่วน โดยจะไม่ถูกตัดบางส่วนออกไปก่อนที่จะมันจะไปถึงปลายทาง

The presentation layer

ชั้นที่ 6 จะทำให้แน่ใจว่า ข้อมูลข่าวสารที่ส่งไปนั้น อยู่ในรูปที่ ปลายทางสามารถรับได้โดยสามารถเข้าใจข้อความและนำไปใช้ได้ด้วย form และภาษาของข้อมูลนั้นจะถูกตัดสินใจโดยแต่ละกลุ่มของ 2 การสื่อสารข้อมูล หน้าที่ของ layer นี้จะมีหน้าที่แปลข้อความต่างๆถ้ามีความจำเป็น ต้องใช้ตัวอย่างเช่น ที่ปลายทางข้างหนึ่งของระบบ จะอ่านข้อมูลต่างๆโดยใช้ ASCII code แต่อีกด้านหนึ่งของ network นั้นใช้ IBM's EBCDIC ดังนั้น layer ชั้นนี้จะทำหน้าที่ จัดการแปลจาก ASCII code ไปเป็น IBM's EBCDIC หรือจาก IBM's EBCDIC ไปเป็น ASCII code

The application layer

ชั้นที่สูงสุดของ model นี้ คือ application layer ชั้นที่ 7 นี้ จะจัดการเกี่ยวกับข้อมูลข่าวสารทั้งหมดในหลายๆแบบด้วยกัน มันจะจัดการแหล่งที่มาสำหรับการย้าย file ของข้อมูลข่าวสารกระจายผลที่ได้ออกไปและแปล หรือย้ายข้อความที่สมบูรณ์แล้วไปยังอีก program ที่ user (ซึ่งนั่งอยู่เหนือ layer นี้) ต้องการที่จะใช้ข้อมูลข่าวสารนั้น

การสื่อสารในระบบ LAN จะใช้การสื่อสารแบบ packet จะเกี่ยวข้องกับ 2 ระดับล่างซึ่งได้แก่

1. ระดับฟิสิกส์คัล (physical layer) เป็นระดับที่เกี่ยวข้องกับการรับข้อมูลเป็นบิต และทางกลของอุปกรณ์ที่เกี่ยวข้องกับการเชื่อมต่อทำให้เกิดการสื่อสารได้
2. ระดับดาต้าลิงค์ (data link layer) เป็นระดับทำการแปลงการรับส่งข้อมูลที่มีความไม่แน่นอนเกิดขึ้น โดยการจัดรูปแบบข้อมูลเป็น บล็อก (block) เป็นเฟรม (frame) พร้อมทั้งมีการตรวจสอบข้อผิดพลาด เพื่อจะได้รับรู้ว่าข้อมูลที่ได้รับมาถูกต้องหรือไม่

จากที่กล่าวทั้งหมด จะพบว่า ในระบบ LAN ที่ลักษณะเฉพาะที่สำคัญ 2 ประการ คือ

ประการแรก ข้อมูลที่จะส่งออกไปได้ จะมีการกำหนดเป็น frame ที่มี address ที่แน่นอน

ประการที่สอง ข้อมูลที่ส่งออกไปไม่มีการเลือกเส้นทางในเน็ตเวิร์ค

จากข้อกำหนดของ IEEE 802 ได้จัดแบ่งระดับของ LAN ขึ้นใหม่เป็นการแยกระดับออกเป็นดังรูป โดยแบ่งเป็น 3 ระดับ ซึ่งจะเทียบได้กับ OSI 2 ระดับโดยส่วนบนของระดับที่กำหนด

คือ LLC (Logical Link Control) มีหน้าที่สำคัญคือ

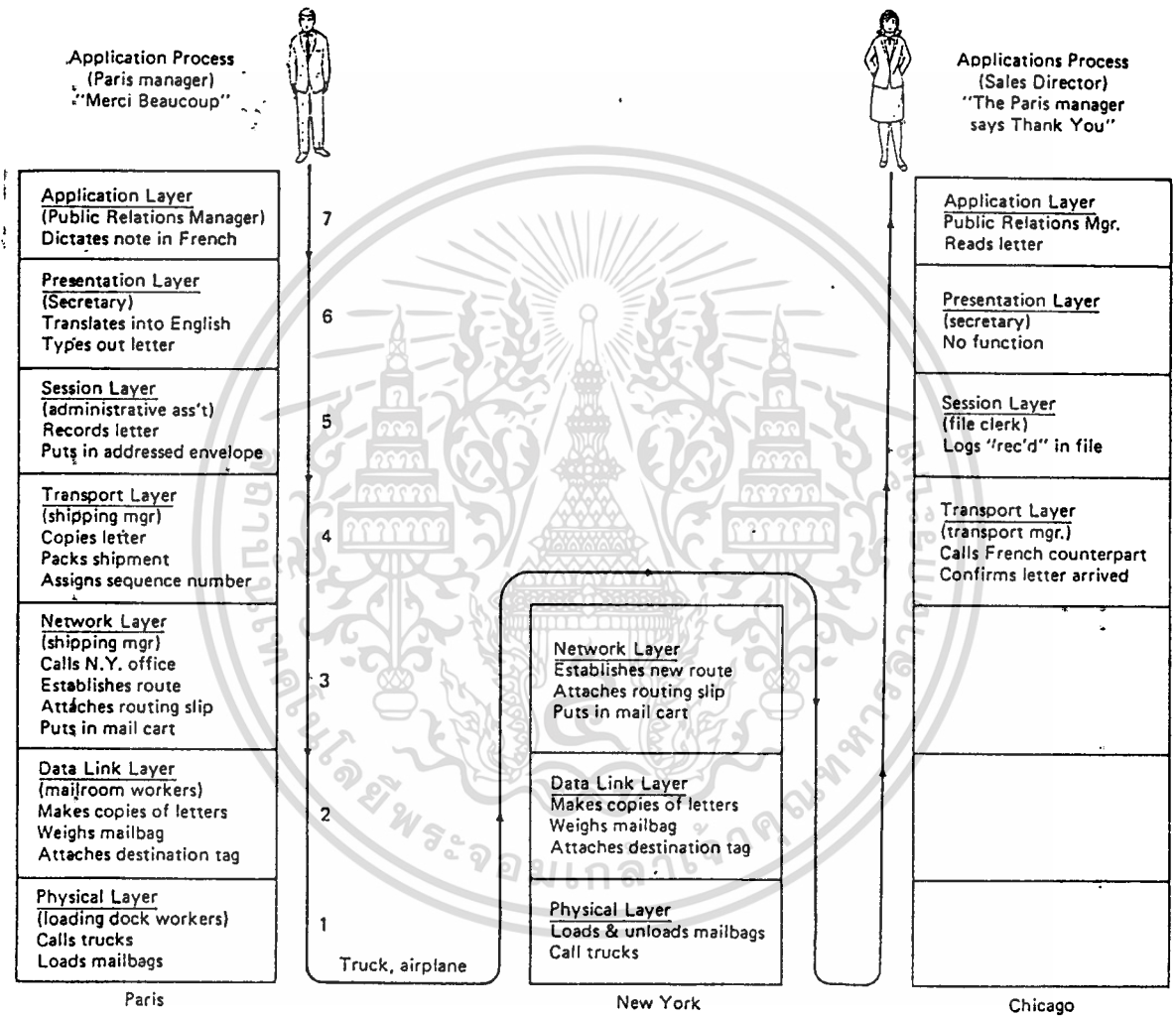
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีจุดเชื่อมต่อที่เรียกว่า SAP (Service Access Point) เพื่อให้เชื่อมต่อกับระดับที่อยู่ข้างเคียงได้
- ในระบบการส่งข้อมูล จะจัดข้อมูลเป็น frame มีการกำหนดแอดเดรสและฟิลด์ CRC
- ในการรับข้อมูล จะแยกข้อมูลออก frame เพื่อตรวจสอบแอดเดรสและตรวจ CRC สำหรับในระดับ MAC (Medium Access Control) ซึ่งอยู่ในระดับล่างถัดลงมา เมื่อเทียบกับ OSI ก็ยังอยู่ในระดับต่ำลึกลงนั้น จะทำงานในหน้าที่ต่อไปนี้
- เป็นการจัดการสำหรับการ Access ข้อมูลที่มาจากต้นทางหลายแห่ง และไปยังปลายทางได้หลายที่
- ทำหน้าที่บางส่วนคล้ายระดับ LLC
- เอนโค้ด และดีโค้ดสัญญาณ
- ทำหน้าที่ชิง โครโนซ์ สร้างสัญญาณ
- ส่งและรับปิดข้อมูล

ที่กล่าวมาทั้งหมดนี้เห็นว่ายากซับซ้อนมาก แต่ก็ผู้สร้างและนำมาใช้ ตัวอย่างที่จะกล่าวถึงนี้ จะแสดงให้เห็นว่า ทั้ง 7 layers ของ OSI model นี้จะสามารถนำไปใช้ระบบการสื่อสารข้อมูลในความเป็นจริงได้อย่างไร สมมติว่าผู้จัดการของ office ที่ Paris ของ บริษัท New York ต้องการที่จะส่งข้อความ thank you ถึง sales director ของอีกบริษัทหนึ่งใน Chicago ตามตัวอย่างข้างล่างนี้จะแสดงวิธีที่ OSI model จะทำหน้าที่อย่างไร ดังรูปที่ 2.32

2.4.1 ชนิดของ Protocal

topology นั้นใช้เป็นส่วนประกอบส่วนหนึ่งของ network ที่ให้จัดระบบให้ network ทำหน้าที่อย่างเหมาะสมเท่านั้น แต่ในระบบนั้นมี user อยู่มากมายดังนั้น protocal จึงจำเป็นที่จะต้องถูกติดตั้ง เข้าไปในระบบเพื่อที่จะเป็นตัวกำหนดว่าใครจะเป็นคู่ที่ติดต่อกัน ใครเป็นผู้พูดภายใต้เงื่อนไขต่างๆและจะเป็นเช่นไรถ้ามี 2 user ต้องการที่จะส่งข้อมูลภายในเวลาเดียวกันหรือใกล้เคียงกันส่วนพวกนี้ protocal ต้องเป็นผู้จัดการทั้งหมดมี protocal อยู่หลายระดับที่เราจะใช้บ่อยคือ command/response, interrupt-driven, token passing และ collision

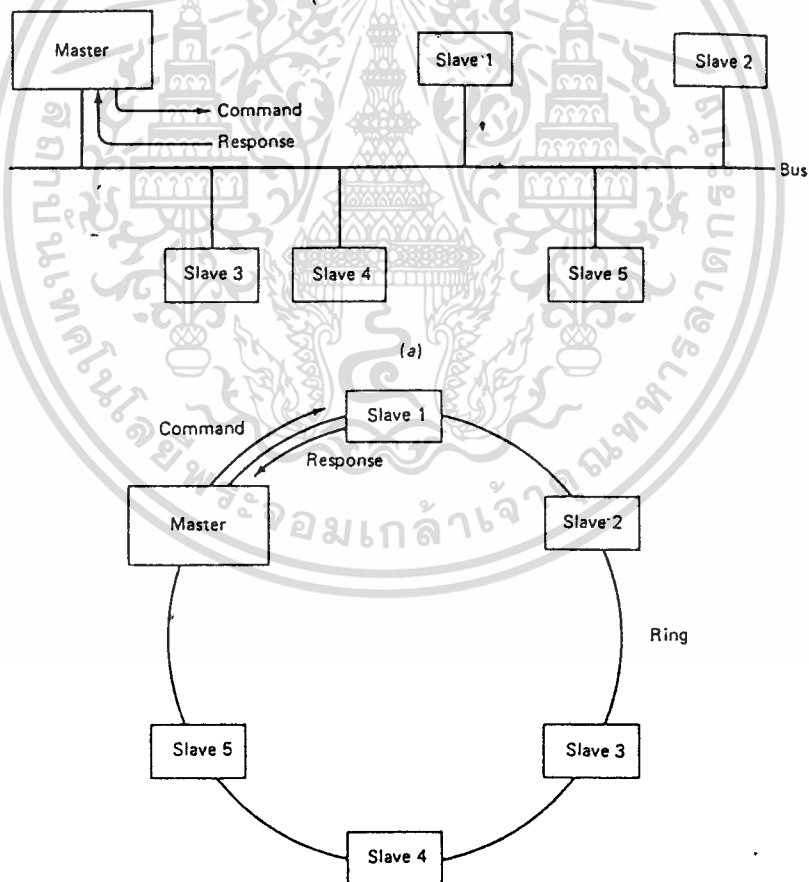


รูปที่ 2.32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COMMAND/RESPONSE

ในระบบ command/response นั้นจะมี user หนึ่งซึ่งเราเรียกว่า master และ user อื่นทั้งหมดในระบบเราว่า slave ตัว slave นี้จะสามารถส่งข้อความได้ก็ต่อเมื่อ master ส่งคำสั่งไปบอกให้ทำ เช่นนั้นจึงจะทำได้ สรุปแล้วหลักการพื้นฐานของ protocol นี้ก็คือ master ออกคำสั่งและ slave เป็นผู้ตอบนั่นเอง เมื่อไหร่ก็ตามที่ user ต้องการที่จะส่งข้อมูลไปยัง master นั้น ตัวมันจะต้องรอจนกว่ามันจะสามารถทำได้โดยจะได้รับคำสั่งจาก master แต่ถ้า



รูปที่ 2.33 command/reponse protocol on bus and ring

master ต้องการที่จะส่งข้อความไปยัง slave นั้นก็สามารถทำได้โดยรวมข้อความนั้นเข้าไปกับคำสั่งและส่งไปยัง slave จากนั้นจึงรอคำตอบจาก slave เพื่อที่จะดูว่า slave ได้รับข้อความนั้นเรียบร้อยแล้วหรือยัง

ดังตัวอย่างรูปที่ 2.33 มี master และ slave 5 ตัวอยู่ในระบบ network ในการที่จะรับข้อมูลมาจาก slave ตัวที่ 3 master จะสั่งว่า "slave 3 นำข้อมูลมาให้แก่ master " slave ก็จะตอบกลับไปยัง master พร้อมกับส่งข้อความนั้นออกไปพร้อมกับคำตอบด้วย แต่ถ้าจะเป็นข้อความไปยัง slave ตัวที่ 4 master จะสั่งว่า "slave 4 นี่คือการส่งที่ต้องการจะส่งให้" slave ตัวที่ 4 ก็จะรับข้อมูลเอาไว้และตอบกลับไปยัง master ว่า "slave 4 ได้รับข้อความเป็นที่เรียบร้อยแล้ว"

slave ตัวหนึ่งๆ นั้นอาจจะส่งข้อความไปยัง slave ตัวอื่นๆก็ได้โดยการใช้ command /response protocol แต่จะเป็นการกระทำโดยทางอ้อม โดยครั้งแรก slave จะส่งข้อมูลไปยัง master และต่อจากนั้น master ก็จะส่งข้อมูลนั้นไปยัง slave ตัวที่ต้องการ ดังเช่น

master : " slave 2 นำข้อมูลมาให้แก่ master "

slave 2 : " นี่คือการส่งสำหรับ slave 1 "(จะไปพร้อมกับข้อมูล)

master : " slave 1 นี่คือการส่งที่ส่งมาให้แก่ slave 1 "(จะไปพร้อมกับข้อมูล)

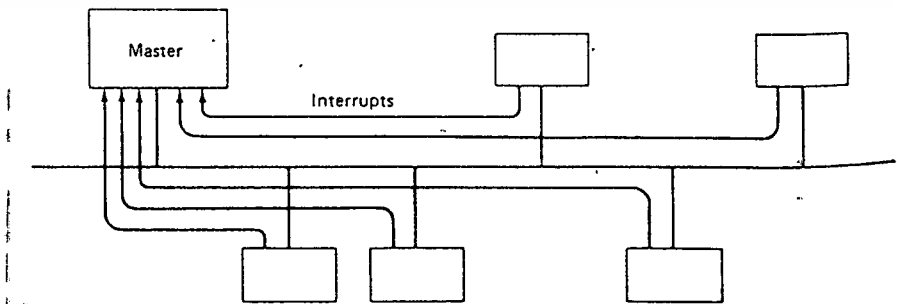
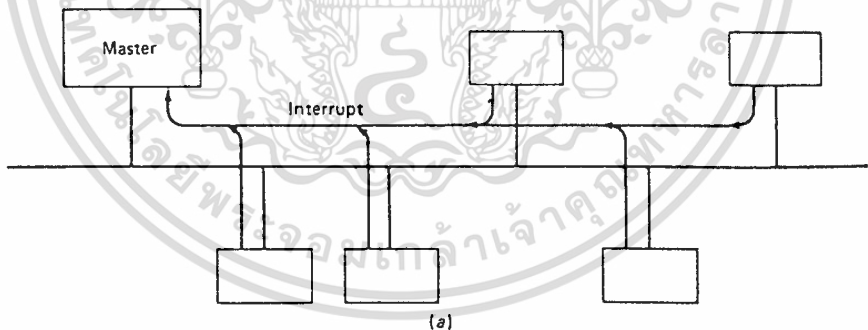
slave 1 : " slave 1 ได้รับข้อมูลเป็นที่เรียบร้อยแล้ว "

ถึงแม้ว่า protocol นี้จะดูว่าง่ายต่อการใช้แต่มันก็ไม่มีประสิทธิภาพเพียงพอ เพราะว่า master จะต้องใช้เวลาอย่างมากที่ใช้ในการตรวจสอบดู slave ทุกๆ ตัว แม้ว่าในบางครั้งการส่งข้อมูลนั้นจะมีข้อมูลเป็นจำนวนน้อยมากแต่การที่จะส่งข้อมูลจาก slave ไปยัง master นั้นต้องใช้เวลาในการส่งข้อความถึง 2 ข้อความ (command and response) ดังนั้นระบบ network จึงเสียเวลามากในการที่จะส่งข้อความเหล่านั้น และถ้าจะส่งข้อมูลจาก slave หนึ่งไปยังอีก slave หนึ่งจะต้องใช้ข้อความในการส่งถึง 4 ข้อความทำให้ระบบ network นี้เสียเวลามากขึ้นไปอีก

INTERRUPT-DRIVEN

interrupt-driven protocol นี้ได้ถูกพัฒนาจาก command/response protocol โดยที่จุดประสงค์ของ interrupt-driven คือพัฒนาประสิทธิภาพในการทำงานของ network ซึ่งเป็นจุดอ่อนของ command/response protocol

ใน interrupt-driven protocol นี้จะมี master เช่นเดียวกับใน command/response protocol แต่แทนที่ master จะทำการ polling (ตรวจสอบ slave อยู่ตลอดเวลา) slave แต่ละตัวอยู่ตลอดเวลาสำหรับข้อมูลที่มีการส่ง master จะคอยฟังสัญญาณพิเศษซึ่งเรียกว่า interrupt โดยที่ slave จะเป็นผู้ส่งบอกให้ master ทราบว่า ขณะนั้น slave ต้องการที่จะส่งข้อมูลแล้ว ดังรูปที่ 2.34 เมื่อเกิดการ interrupt ขึ้น master จะส่งข้อความไปพูดกับ slave ตัวที่ส่งการ interrupt เข้ามาว่า "คุณมีข้อมูลที่จะส่งมาให้ master หรือไม่" slave ตัวนั้นก็จะตอบกลับไปว่า "ไม่มีข้อมูลที่จะส่ง" หรือส่งข้อมูลนั้นกลับไปยัง master โดยการกระทำตามวิธีการนี้จะมีประสิทธิภาพมากกว่า วิธีการที่จะต้องมาคอยตรวจสอบอยู่ตลอด



เวลามี slave ใดบ้างที่ ต้องการจะส่งข้อมูลซึ่งเป็นวิธีการที่เราได้กล่าวมาแล้วใน command /response ซึ่งวิธีนี้ ถ้าไม่มีข้อมูลที่จะส่งก็จะทำให้เสียเวลาโดยเปล่าประโยชน์ ที่จะต้องคอยไปตรวจสอบอยู่เสมอ

การที่เราจะพัฒนาระบบนี้สามารถทำได้โดยใช้สายสัญญาณที่จะมาทำเป็นสาย interrupt มากขึ้นโดยสาย 1 เส้นจะสำหรับ 1 slave โดยเชื่อมจาก slave ทุกๆ ตัวไปยัง master แต่ละ slave ก็จะสามารถส่งสัญญาณ interrupt ของมันออกไปยัง master ได้โดยจะไม่ยุ่งเกี่ยวกับการ interrupt ของ slave อื่นๆ เมื่อเกิดการ interrupt ขึ้น slave ตัวที่ส่งสัญญาณออกไปนั้นไม่เพียงแต่ที่จะส่งสัญญาณ interrupt ออกไปเท่านั้นแต่ยังส่งข้อความออกไปบอกว่า slave ตัวใดเป็นตัวที่ต้องการขอ interrupt เพื่อที่จะให้ master ทราบ master จึงจะส่งสัญญาณตอบไปยัง slave ได้ถูกต้องว่าเป็น slave ตัวใด

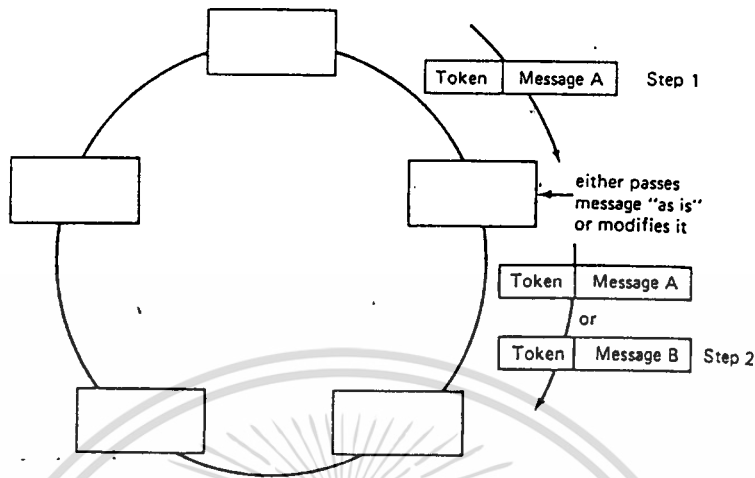
วิธีการ interrupt-driven นี้จะประหยัดเวลามากเพราะ master จะตรวจสอบไปยัง slave ตัวที่ต้องการ interrupt เข้ามาเท่านั้นซึ่งจะทำให้ไม่เสียเวลาจากการที่จะต้องไปตรวจสอบทุกๆ user node หรือ slave ว่า มีข้อมูลที่ต้องการที่จะส่งหรือไม่ ทำให้วิธี interrupt-driven นี้มีประสิทธิภาพมากในการนำไปใช้ในระบบ network

TOKEN PASSING

วิธีที่ใช้ master/slave นั้นเราใช้กันใน command/response หรือใน interrupt-driven protocol ซึ่งมีข้อเสียอยู่บ้างที่ทำให้ระบบนี้ไม่มีประสิทธิภาพเท่าที่ควรสำหรับการใช้งานบางประเภทโดยจะมีประสิทธิภาพต่ำหรืออาจจะมีปัญหาเมื่อข่าวสารต่างๆ ต้องการที่จะส่งไปที่ user อื่นที่เป็น slave แต่ไม่ใช่ master

ใน command/response และ ระบบ master/slave นั้นเราต้องการใช้ network ที่มีความเร็วสูง เพื่อที่จะใช้ในการส่งข้อความมาทุกๆ slave ไปยังทุกๆ slave หรือจาก slave หนึ่งไปยังอีกหลายๆ slave ทางเส้นทางหนึ่งที่จะไม่ใช่ master/slave คือการใช้ token passing protocol ใน token-passing นั้นจะไม่มี user ใดที่เป็น master และ user อื่นๆ ก็จะไม่เป็น slaves ทุกๆ user จะมีความสัมพันธ์กันหมดและมีศักดิ์ศรีเท่ากัน หรือ อำนาจ

ในการกระทำการเท่ากันหมด ถึงแม้ว่าในบางเหตุการณ์ อาจจะมีบางหน่วยที่ทำหน้าที่เป็นตัวร่วมก็
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.35 token-passing

Token-passing จะทำหน้าที่ดังนี้ คือ ข้อความซึ่งจะถูกจัดรูปแบบเป็นแบบพิเศษซึ่งเราเรียกว่า token จะถูกส่งจาก user node หนึ่งๆ ไปยัง user node อื่นๆ ในทิศทางใดทิศทางหนึ่ง ดังรูปที่ 2.35 ที่ node ที่ได้รับข้อความ node ถัดไปนั้น user จะตรวจสอบ token ที่ได้มานี้ แต่ละ user node จะได้เวลาจำกัดอยู่เวลาหนึ่งในระหว่างเวลานี้เอง user สามารถจะเอา token ออกมาถ้า token อันนี้ บอกว่าเป็นของ user นี้ หรือว่าจะใส่ข้อความเพิ่มเข้าไปใน token นี้ก็ได้และในช่วงเวลานี้ node อื่นๆ ที่อยู่ใน network นี้จะทำหน้าที่คอยฟังสัญญาณใน network ต่อไปเรื่อยๆ จนกว่าจะได้ยินสัญญาณ วิธีการ token passing นี้จะดำเนินไปเรื่อยๆ จนกระทั่ง node ที่เป็นตัวที่ส่งข้อความออกไปนั้นได้รับ token คืนกลับมาจาก node สุดท้ายใน network นี้ ซึ่งจะทำให้ node ที่เป็นตัวส่งข้อความรู้ว่า node ที่เป็นตัวรับข้อความหรือ node ที่มันต้องการจะส่งข้อความให้มันได้รับข้อความเรียบร้อยแล้ว

ผลของการใช้ token passing นี้คือ แต่ละ user node จะควบคุมการส่งข้อมูลบน network ได้เวลาใช้เวลาไปเท่าไร node ที่มี token อยู่ด้วยก็จะสามารถควบคุมได้ในช่วงระยะเวลาหนึ่งเป็นชั่วคราว แล้ว token อยู่ด้วยก็จะสามารถควบคุมได้ในช่วงระยะเวลาหนึ่งเป็นการชั่วคราว แล้ว token นี้ก็ผ่านไปเรื่อยๆ จาก node หนึ่งไปยังอีก node หนึ่งไปยังอีก node หนึ่งตามลำดับ ทำให้แต่ละ node นั้นมีช่วงเวลาที่จะควบคุม token นี้

เอกสารนี้เป็นเอกสารลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถนำไปใช้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประโยชน์ที่ได้อย่างมากจากการใช้ token passing นี้คือ แต่ละ user จะเข้าไปใน network ได้ในเวลาที่ถูกกำหนดไว้แล้ว สิ่งนี้ เป็นสิ่งที่สำคัญมากในการที่เราสามารถที่จะกำหนด เวลาได้อย่างแน่นอน และ เป็นเวลาที่น้อย สำหรับการที่จะนำไปใช้งาน เวลา รวมทั้งหมดที่เป็น เวลาที่นานที่สุดในระบบ network นี้ (เวลาเมื่อ user node หนึ่ง ได้ปฏิบัติการแล้วได้กลับมา ทำซ้ำอีกครั้งหนึ่ง) จะเท่ากับเวลาที่แต่ละ node จะได้ token ซึ่งจะเป็นไปตาม

$$\text{จำนวนของ nodes} * \text{เวลาในแต่ละ node}$$

เวลาสำหรับที่ข้อความเดินทาง ไประหว่าง node จะต้องรวมเข้าไปในเวลารวมทั้งระบบ network ด้วย แต่โดยปกติแล้ว เวลานี้จะเป็นเวลาที่น้อยมากเมื่อเทียบกับเวลาที่ token ของ แต่ละ node แต่ถ้าเวลานี้เป็นเวลาที่ไม่น้อยนักเท่าไรเราจะต้องคิดเข้าไปด้วยโดยจะเป็นไปตาม

$$\text{จำนวนของ node} * (\text{เวลาในแต่ละ node} + \text{เวลาที่เดินทางไประหว่างแต่ละ node})$$

Token passing จะรับประกันว่ามันจะ ไม่มีความผิดพลาดที่จะเกิดขึ้นในระบบ network นี้ เพราะแต่ละ user จะทำการส่งข้อมูลนั้นจะทำได้ก็ต่อเมื่อมันมี token อยู่เท่านั้นทุกๆ user นั้น จะต้องคอยฟังอยู่เรื่อยๆ เมื่อมันไม่มี token และ ต้องคอยเตรียมตัวให้พร้อมเพื่อที่จะรับ token เมื่อ token ผ่านมาถึงมัน อย่างไรก็ตาม token passing ก็ยังมี ข้อเสียที่เป็นพื้นฐานอยู่อย่าง หนึ่ง คือ ถ้ามี node ใด node หนึ่ง เสียหรือไม่ทำงานจะทำให้ token ไม่สามารถผ่านไปได้ และ ทำให้ระบบ network นี้เสียไปวิธีทางแก้ไขของปัญหาเหล่านี้คือใน token passing เรา ต้องสร้างกฎเกณฑ์ พิเศษออกมาบางอย่าง และ ต้องใส่เข้าไปใน token passing โดยให้มัน check ตัวเองอย่าง automatically คือเมื่อ node ใด node หนึ่งไม่ได้รับ token มาเป็น

เวลาหนึ่ง ซึ่งเป็นเวลาที่เรารั้งเวลาเอาไว้เวลาหนึ่ง เช่น สมมติเรารั้งว่า เวลาที่มากที่สุดเป็น ไม่ว่าการนี้โดยทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

100ms แล้วถ้า node ใด node หนึ่งไม่ได้รับ token มาเป็นเวลา 100ms แล้วแสดงว่าระบบ network นี้มีปัญหา token ไม่สามารถที่จะผ่านไปตามทางได้ ดังนั้น network นี้จะทำการ restarted ระบบใหม่ และทำงานต่อแต่เวลาที่เรารั้งเอาไว้ จะต้องเป็นเวลาที่เหมาะสม เพื่อไม่ให้มีการเริ่มต้นใหม่ โดยไม่จำเป็น

CSMA/CD

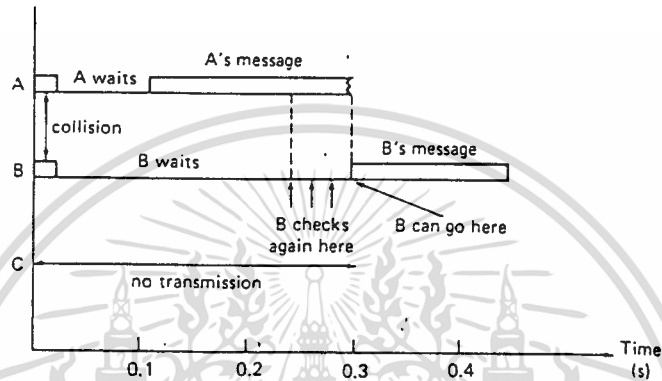
network protocol อันสุดท้ายที่เราจะกล่าวถึงก็คือ carrier sense multiple access/collision detect(CSMA/CD) แนวความคิดของ CSMA/CD นั้นแตกต่างจากแนวความคิดของ token passing มาก โดยกฎเกณฑ์ของ CSMA/CD คือ node ใดๆ ก็ได้ใน network สามารถทำการส่งข้อความได้ทั้งหมด หลังจากที่ฟังให้แน่ใจแล้วว่า ระบบ network ตอนนี้อย่าง หรือไม่มีใครใช้ network อยู่ แต่ถ้า network นี้ไม่ว่าง เพราะ node อื่นได้ทำการส่งข้อมูลกันอยู่ node ที่จะส่งนั้นจะทำการรอไปจนกระทั่งระบบ network นี้ว่างแล้ว จึงทำการส่งข้อมูลได้โดยจะแตกต่างจาก token passing คือ token passing นั้นจะมี node ที่ส่งข้อความได้เพียง node เดียวเท่านั้น คือ node ที่มี token แต่ใน CSMA/CD นั้น node ใดจะส่งข้อมูลก็ได้ถ้าระบบ network นั้นว่างอยู่

ดูๆ แล้วระบบนี้น่าจะเป็นระบบที่ง่าย แต่มันก็ไม่ง่ายอย่างที่คิดเพราะว่ามันยังมีโอกาสที่จะเกิดปัญหาได้ เมื่อมี node อื่นๆ ที่อยากจะส่งข้อมูล หรือทำสิ่งๆ ที่เหมือนกันในเวลาเดียวกัน โดยตรวจสอบระบบว่า network ตอนนี้อย่างหรือเปล่า ถ้าว่างมันก็จะส่งข้อมูลลงไปในระบบ ในช่วงเวลาที่ระบบว่างนี้ node หนึ่งก็เริ่มที่จะส่งข้อมูลออกไป และอีก user node หนึ่งก็ ตรวจสอบระบบเหมือนกันและก็พบว่าระบบว่างเหมือนกันจึงเริ่มที่จะส่งข้อมูลออกไป พร้อมกันผลที่เกิดขึ้นก็คือ เกิดการชนกัน (collision) ของข้อมูล 2 ชุด

CSMA/CD ได้รู้ถึงปัญหาที่จะเกิดขึ้นแบบนี้เรียบร้อยแล้ว และได้เตรียมการพร้อมที่จะแก้ไข แล้วคือเมื่อเกิดการชนกันของข้อมูลจะเป็นผลทำให้ระดับสัญญาณในระบบ network เปลี่ยนไปโดยในแต่ละ node นั้นจะมีวงจรอย่างหนึ่งที่คอยตรวจสอบระดับสัญญาณนี้ (detect) ถ้ามันมีการเปลี่ยนแปลงทันทีที่ดังนั้น แต่ละ node ที่ส่งข้อมูลออกไปก็จะทราบได้ว่าเกิดการชนกันของข้อมูลขึ้น (ดังนั้น CD จึงมาจาก "collision detect") ต่อจากนั้น แต่ละ user node จะต้องทำตาม

กฎเกณฑ์ ที่เหมือนกันดังต่อไปนี้: หยุดการส่งข้อมูลนั้นทันทีที่ชนกัน นำข้อมูลนั้นกลับมาและ คอย ไม่ว่าการชนกันทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อที่จะพยายามส่งข้อมูลออกไปอีกครั้ง โดยเวลาที่คอยต้องเป็นเวลาที่สุ่มขึ้นมา (random time) ซึ่งแต่ละ user node จะมีเวลาที่สุ่มขึ้นมาไม่เท่ากัน วงจรในแต่ละ node นั้นได้ถูกออกแบบไว้ให้ใช้เวลา random ที่แตกต่างกันออกไป เมื่อเกิดมีการตรวจสอบและจับได้ว่ามีการชนกันเกิดขึ้น



รูปที่ 2.36 กรณีเกิด collision ของ CSMA/CD protocol

ซึ่งแนวความคิด ก็คือ เมื่อ 2 user node ผู้ที่เป็นผู้ส่งข้อมูลออกไปชนกันนั้นจะมีเวลาที่นำข้อมูลกลับมา และเวลาที่คอยไม่พร้อมกัน และต่อจากนั้น node หนึ่งจะกลับมาในระบบก่อนที่อีก node หนึ่งจะกลับมาเพราะเวลาที่สุ่มมานั้นไม่เท่ากันจากนั้น protocol นี้ก็จะเริ่มต่อไปได้โดยไม่มีการชนกัน

เรามาดูตัวอย่างเพื่อที่จะดูว่า protocol นี้ ทำงานได้อย่างไร

สมมติว่า network นี้มี 3 users คือ A, B และ C user A มีข้อมูลที่ต้องการที่จะส่งไปใน network ให้ user C และ user B ก็มีข้อมูลที่จะส่งไปใน network ให้ user A ดังนั้นเริ่มทำงานโดยที่ A ตรวจสอบดูแล้ว network ว่างจึงเริ่มทำการส่งข้อมูลออกไป เช่นเดียวกัน เพราะว่า user B ก็ตรวจสอบที่เวลาเดียวกัน และ เริ่มส่งข้อมูลออกไปเช่นเดียวกัน เพราะว่า user B คิดว่าระบบ network ว่างเช่นเดียวกัน จากนั้น ทั้ง A และ B ตรวจพบการชนกันเกิดขึ้น จึงหยุดการส่งทั้งคู่ user A คอยเป็นเวลา 0.1s และ user B คอยเป็นเวลา 0.2s ก่อนที่จะพยายามเริ่มส่งใหม่อีกครั้ง (ดังรูปที่ 2.36)แน่นอนว่า A รอคอยเป็นเวลาน้อยกว่า user A จึงได้เข้าไปใน network ก่อน user B ถ้า user A ยังคงกระทำการส่งอยู่บน network

หลังจากที่ user B คอยมาเป็นเวลา 0.2s แล้ว user B จะต้องรอจนกระทั่ง A ทำการส่งข้อมูลให้เสร็จสิ้น ถ้า A ทำการส่งเสร็จแล้วจึงเป็นคราวของ user B ที่จะส่งข้อมูล แต่ถ้ามีการชนกันเกิดขึ้นอีกครั้ง A, B และ C ก็จะมีการรอคอยที่แตกต่างกันมากกว่า ที่เคยรอในครั้งแรก

ทำไมจึงต้องใช้ random time แทนที่จะให้มีความเฉลี่ยที่เกิดขึ้นใน network ซึ่งแต่ละ user มีโอกาสที่เท่ากันในการที่จะเข้าไปใน network ถ้าเวลาที่รอคอยนั้นได้ถูกกำหนดไว้แทนที่จะใช้เวลาที่สุ่มขึ้นแล้ว A ก็จะมี priority ที่มากกว่า B อยู่เสมอและ B ก็จะมี priority ที่สูงกว่า C อยู่เสมอ ดังนั้นข้อมูลที่จะส่งไปใน network มีจำนวนมากแล้วมันจะมีโอกาสเป็นไปได้นานที่ C จะเกิดเหตุการณ์ "locked out" และจะไม่ได้โอกาสที่จะเข้าไปใน network และส่งข้อมูลเลยวิธี CSMA นี้สามารถนำไปใช้จริงๆ ได้ และดีมากแต่มีบางคุณลักษณะที่เราควรจะระลึกถึงไว้ ปัญหาหลักที่อาจจะเกิดขึ้นได้คือ ถ้าจำนวน user เพิ่มขึ้นอย่างมากมายแล้วจำนวนของข้อมูลที่จะส่งไปใน network ก็มีมากตามขึ้นไปด้วยจะทำให้เกิดปัญหาของการจราจรของข้อมูลทำให้เสียเวลาเป็นเวลามาก ผลก็คือเมื่อมี node ที่พยายามจะส่งข้อมูลใน network แต่ขณะนั้น network จึงต้องมีการคอยกันแต่ไม่สามารถบอกได้ว่าจะต้องรอคอยเป็นเวลานานเท่าไร ระบบ network จึงว่างลงแล้ว node นั้นจะทำการส่งข้อมูลได้โดยอาจจะเกิดการชนกัน (collision) หลายครั้งซ้ำแล้วซ้ำเล่ากับ user อื่นๆ ได้

บทที่ 3 ระบบงานทะเบียนนักศึกษา

3.1 รายละเอียดระบบงานเดิม

ระบบงานทะเบียนนักศึกษา ที่ใช้เจ้าหน้าที่ดำเนินงาน ประกอบด้วยระบบงาน 4 งาน มีรายละเอียดดังนี้

- ระบบงานทะเบียนประวัติ
- ระบบงานทะเบียน
- ระบบงานประมวลผลการศึกษา
- ระบบงานจัดทำตารางสอนและตารางสอบ

3.1.1 ระบบงานทะเบียนประวัติ

มีงาน 2 ส่วน

- งานทะเบียนประวัตินักศึกษา
- งานออกใบรับรองและหนังสือสำคัญต่างๆ

1.1 งานทะเบียนประวัตินักศึกษา

มีงานในหน้าที่คือ

- รับขึ้นทะเบียนนักศึกษาใหม่ เก็บประวัติ
- ดูแลเก็บประวัติส่วนตัว และ ผลการศึกษาทุกภาค
- ตรวจสอบคุณสมบัติของผู้สำเร็จการศึกษาตามที่หน่วยงานเอกชน, ราชการขอมา
- ตรวจสอบสภาพนักศึกษาเมื่อหมดภาคการศึกษาทุกภาค ดำเนินการกับนักศึกษาที่หมดสภาพ
- เรื่องลาพัก ลาออก สักพัก และการขอคืนสภาพนักศึกษา
- เปลี่ยนแปลงข้อมูลในประวัติ
- ตรวจสอบข้อมูลต่างๆ เช่น ประวัติส่วนตัวของนักศึกษาที่ยื่นคำร้องขออนุมัติ
ปริญญา
- จัดทำสถิติจำนวนนักศึกษา ในแต่ละภาควิชา แต่ละชั้นปี

1.2 ด้านออกใบรับรองและหนังสือสำคัญต่างๆ

1.2.1 สำหรับผู้ที่กำลังศึกษาอยู่

- บัตรประจำตัวนักศึกษา
- ใบรับรองว่านักศึกษาเป็นนักศึกษาที่กำลังศึกษาอยู่จริง
- ใบรับรองคาดว่าจะสำเร็จการศึกษาจริง
- ใบรับรองรายละเอียดของผู้กำลังศึกษาอยู่

1.2.2 สำหรับผู้สำเร็จการศึกษา

- ใบประมวลผลการศึกษา
- ใบรับรองหรือหนังสือรับรองการสำเร็จการศึกษา
- ปริญญาบัตร
- ใบแทนปริญญาบัตร
- ใบแปลปริญญาบัตร

1.2.3 สำหรับผู้ที่พ้นสภาพนักศึกษา ที่ยื่นคำร้องขอ ที่ถูกถอนชื่อออก ส่งให้ออก

- ใบประมวลผลการศึกษาของแต่ละภาคการศึกษา

3.1.2 ระบบงานทะเบียน

- การลงทะเบียนปกติ
- การลงทะเบียนสาย
- การเพิ่มรายวิชา
- การถอนรายวิชา

ขั้นตอนการปฏิบัติงาน

2.1 ตรวจสอบสภาพนักศึกษา

- มีสิทธิลงทะเบียน
- เตรียมรายชื่อนักศึกษาที่จะลงทะเบียน แยกตามภาค ปีการศึกษาที่เข้าศึกษา
- มีการรับนักศึกษาเข้าใหม่ให้เตรียมรายชื่อนักศึกษาใหม่ในการลงทะเบียน
- ตัดรายชื่อนักศึกษาผู้ที่ไม่ลาออกหรือ ถูกถอนชื่อออกตามคำสั่งของสถาบันการศึกษา
ออกจากบัญชีรายชื่อนักศึกษา โดยประสานงานกับฝ่ายทะเบียนประวัตินักศึกษา

- สำหรับรายชื่อนักศึกษาที่ลาพักหรือถูกสั่งพักการศึกษาตรวจสอบว่า มีสิทธิในการลงทะเบียนในภาคการศึกษานี้หรือไม่
- จัดทำรายชื่อนักศึกษา ผู้ที่มีสิทธิลงทะเบียนประจำภาคนี้
- แก่ไรท์สและชื่ออาจารย์ที่ปรึกษาตามที่คณะหรือสาขาวิชาแจ้งมา

2.2 การลงทะเบียนวิชาเรียน

2.2.1 การเตรียมการลงทะเบียนวิชาเรียน

2.2.2 การดำเนินงานลงทะเบียน

- ตรวจสอบความถูกต้องของเลขทะเบียนนักศึกษา ชื่อ นามสกุล รหัส ชื่ออาจารย์ที่ปรึกษา ภาคการศึกษา ปีการศึกษา รหัสคณะ สาขาวิชา ลายมือชื่ออาจารย์ที่ปรึกษา ลายมือชื่อนักศึกษา ที่อยู่ของนักศึกษาในบัตรลงทะเบียนวิชาเรียน
- ชื่อรหัส จำนวนหน่วยกิต ตรงตามที่กำหนด
- รวมหน่วยกิต และคิดเงินค่าธรรมเนียมการศึกษาประสานกับการเงิน
- รับหลักฐานการลงทะเบียนวิชาเรียนจากนักศึกษา ภายหลังจากการชำระเงินแล้ว
- เก็บหลักฐานการลงทะเบียนวิชาเรียนทั้งหมด
- นำเอกสารการลงทะเบียนทั้งหมด มาตรวจสอบอีกครั้งหนึ่ง เกี่ยวกับสภาพของนักศึกษาว่าถูกต้องหรือไม่ นักศึกษาลงทะเบียนเรียนถูกต้องหรือไม่ นักศึกษาลงทะเบียนเรียนถูกต้องตามรายวิชาที่เปิดสอน ตามโปรแกรมการศึกษา ในแต่ละภาคหรือไม่ จำนวนหน่วยกิตรวมที่นักศึกษาลงทะเบียน ถูกต้องตามสภาพของนักศึกษาหรือไม่ ถ้าพบที่ผิดพลาดก็ดำเนินการแก้ไข
- หลังจากหมดกำหนดการลงทะเบียนสายแล้ว ให้ทำการตรวจสอบรายชื่อผู้ที่ไม่มาลงทะเบียนวิชาเรียน โดยมีได้ลาพัก รวบรวมรายชื่อเหล่านี้เสนอฝ่ายทะเบียนประวัติ ทำการตรวจสอบและดำเนินการให้เป็นไปตามกฎของสถาบันการศึกษาต่อไป

- แจ้างผลการลงทะเบียนให้นักศึกษา อาจารย์ที่ปรึกษาสาขาวิชา และอาจารย์ผู้สอนทราบ โดยทำใบแจ้งผลดังต่อไปนี้
- จำนวนและรายชื่อ นักศึกษา แยกตามรายวิชา
- รายงานผลการลงทะเบียนรายบุคคล
- รายชื่อนักศึกษาที่ไม่มาลงทะเบียน ของแต่ละคณะหรือสาขาวิชา

2.3 การเพิ่มและถอนรายวิชา

1. แจกบัตรเพิ่มวิชาเรียน หรือ ถอนวิชาเรียน

2. ดำเนินการเพิ่ม-ถอน หรือ เปลี่ยนวิชาเรียน

- ตรวจสอบความถูกต้องทุกอย่าง
- ชื่อ รหัส หน่วยกิต ว่าถูกต้องหรือไม่
- ตรวจสอบการลงทะเบียนให้เป็นไปตามกฎหรือข้อบังคับของสถาบันฯ
- การคิดเงินค่าธรรมเนียม ในการเพิ่มอย่างเดียว หรือเพิ่มและถอนทั้ง 2 อย่าง
- รับหลักฐานการเพิ่มหรือถอน หลังจากชำระเงินแล้ว
- รวบรวมรายชื่อนักศึกษาที่เพิ่มหรือลดวิชาแยกตามภาคให้กับอาจารย์ผู้สอน
- ติดต่อให้นักศึกษามาแก้ไขถ้ามีการลงผิดพลาด
- รวบรวมคำร้องขอคืนเงิน เช่น ปิดวิชา
- เก็บหลักฐานการลงทะเบียน
- รายงานผลการศึกษารายบุคคล ให้นักศึกษาทราบ หลังเสร็จสิ้นระยะเวลาการเพิ่ม-ถอนวิชาแล้ว
- ทำรายงานรหัส และรายชื่อนักศึกษาลงทะเบียนแยกตามรายวิชา

3. ระบบงานการประมวลผลการศึกษา

หน้าที่และความรับผิดชอบ

- ทำบันทึกแจ้งอาจารย์ผู้สอนให้ทราบวันที่จะต้องส่งผลสอบที่สมบูรณ์มายังฝ่ายประมวล
- รับผลสอบจากอาจารย์ผู้สอน

- เก็บรักษาต้นฉบับผลสอบ
- รวบรวมผลสอบที่ตรวจสอบแล้ว คำนวณคะแนนเฉลี่ย คะแนนเฉลี่ยสะสม จัดทำ
ใบรายงานผลการศึกษารายบุคคล
- แจงผลให้นักศึกษาทราบ
- รวบรวมใบรายงานผลการศึกษารายบุคคลส่งให้ภาค ส่งให้อาจารย์ที่ปรึกษาเก็บ
เป็นหลักฐาน
- รวบรวมใบรายงานผลการศึกษารายบุคคล ส่งฝ่ายทะเบียนประวัติ เก็บเป็นประวัติ
และเพื่อใช้เป็นใบประมวลผลการศึกษา

4. ระบบงานจัดทำตารางสอน ตารางสอบ

ประกอบด้วยงาน

- จัดทำตารางสอน ตารางสอบ
- เปลี่ยนแปลงตารางสอน ตารางสอบเมื่อมีการยื่นคำร้อง
- แจงให้อาจารย์ผู้สอนทราบ

จากที่กล่าวมาข้างต้น จะเห็นได้ว่าระบบมีความซับซ้อนมาก การตรวจสอบความถูกต้องทำได้ล่าช้า และอาจเกิดข้อผิดพลาดได้ การนำเอาคอมพิวเตอร์มาช่วยในการบริหารงานทะเบียนโดย
ใช้ข้อมูลร่วมกันเป็นศูนย์กลาง จึงทำให้การบริการแก่นักศึกษาเป็นไปอย่างสะดวก รวดเร็วและ
สามารถประหยัดแรงงานเจ้าหน้าที่ได้มาก

3.2 การบริหารระบบทะเบียนนักศึกษาโดยใช้คอมพิวเตอร์

ในระบบที่พัฒนาขึ้นได้พยายามรักษาระดับขั้นตอนการปฏิบัติงานแบบเดิมไว้ โดยในส่วนที่ติดต่อกับผู้ใช้จะเป็นในลักษณะที่เข้าใจได้ง่าย มีการป้องกันข้อผิดพลาดต่างๆ และลดขั้นตอนการทำงานให้น้อยลง แต่จะมีการเปลี่ยนแปลงในส่วนของการจัดการข้อมูลโดยรวมส่วนต่างๆ ให้เข้าเป็นระบบเดียว เพื่อไม่ให้เกิดการซ้ำซ้อนของการจัดเก็บและการตรวจสอบความถูกต้องของข้อมูล สามารถทำได้อย่างรวดเร็ว และข้อมูลมีความปลอดภัย มีความเชื่อถือได้

ระบบการทำงานที่ออกแบบมา จะแบ่งการทำงานออกเป็นส่วนที่เกี่ยวข้อง 3 ส่วน โดยแต่ละส่วนจะมีส่วนที่เกี่ยวข้องกัน และยังมีการแยกย่อยออกไปอีก ดังจะได้กล่าวต่อไป คือ

1. ส่วนของฝ่ายทะเบียน
2. ส่วนของอาจารย์
3. ส่วนของนักศึกษา

3.2.1 ส่วนของฝ่ายทะเบียน

เป็นส่วนที่ฝ่ายทะเบียนจะเป็นผู้จัดการทั้งหมด โดยจะจัดการทุกๆ ส่วน ที่เกี่ยวกับการลงทะเบียน ระเบียน เกรด และผลการศึกษา โดยมีระบบการทำงานดังนี้

1. ประวัตินักศึกษา

แบ่งออกเป็น

- ประวัติส่วนตัว

ให้เจ้าหน้าที่บันทึกข้อมูลต่างๆ เกี่ยวกับนักศึกษา เช่น ชื่อ นามสกุล ที่อยู่ ชื่อผู้ปกครอง เหล่านี้เป็นต้น สำหรับนักศึกษาใหม่ ส่วนที่เป็นนักศึกษาเก่าจะสามารถแก้ไขประวัติของนักศึกษาได้

- ประวัติการศึกษา

จะเป็นรายละเอียดเกี่ยวกับข้อมูลการศึกษาของนักศึกษา ซึ่งจะสามารภเพิ่มเติมและแก้ไขได้

โดยจะมีระบบการทำงานเป็นดังนี้

- ป้อนข้อมูลรายละเอียด ประวัติของนักศึกษาใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ออกรหัสนักศึกษา โดยจะนำรายชื่อนักศึกษาในแต่ละภาค มาเรียงลำดับแล้วออกรหัสประจำตัวนักศึกษา
- นำข้อมูลเก็บเข้าแฟ้มข้อมูลนักศึกษา ตั้งค่าสถานภาพและระดับชั้นการศึกษา
- มีการแก้ไขหรือเพิ่มเติม ส่วนของประวัติของนักศึกษาได้

2. การบันทึกรายวิชา มีระบบการทำงานดังนี้

- ป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของวิชาบังคับเฉพาะภาคการศึกษา โดยจะมีรายละเอียดว่าในแต่ละภาคการศึกษา จะมีวิชาบังคับอะไรบ้างที่เปิดสอน
- ป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของวิชาเลือกเฉพาะภาคการศึกษา โดยจะมีรายละเอียดว่าในแต่ละภาคการศึกษา จะมีวิชาเลือกอะไรบ้างที่เปิดสอน
- ป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของรายวิชาที่เปิดสอนในเทอมปัจจุบัน โดยจะมีการป้อนรหัสวิชา, ชื่อของอาจารย์ผู้สอนวิชานั้น, ชื่อวันและเวลาที่เรียน และช่วงเวลาที่สอบ ซึ่งในส่วนนี้เจ้าหน้าที่ทะเบียนจะสามารถแก้ไขและเพิ่มเติมได้
- ป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของรายละเอียดของวิชาทั้งหมด โดยมีข้อมูลต่างๆ ดังนี้ รหัสวิชา, ชื่อวิชา, จำนวนหน่วยกิต, ภาคทฤษฎี-ปฏิบัติ, วิชาก่อนหน้า ในส่วนนี้จะป้อนรายละเอียดของวิชาทั้งหมดในหลักสูตรไม่ว่าจะเปิดสอน หรือไม่เปิดสอน ในเทอมนี้ก็ตาม

3. การลงทะเบียนของนักศึกษา

- กำหนดช่วงวันลงทะเบียน โดยป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของช่วงวันลงทะเบียน และช่วงวันสุดท้ายของการเพิ่ม, ก่อนรายวิชา ในแต่ละเทอมนั้นๆ
- สรุปผลการลงทะเบียน โดยจะตรวจสอบรายละเอียดของนักศึกษาแต่ละคนว่า ได้ลงทะเบียนอะไรไปบ้างและถูกต้องหรือไม่
- สรุปผลการเพิ่มหรือถอนรายวิชา จะเป็นการตรวจสอบรายละเอียดต่างๆ ของแต่ละวิชาว่ามีการเพิ่มหรือถอนอะไรไปบ้างในเทอมๆ นั้น

4. ผลการศึกษาของนักศึกษา

- การบันทึกเกรด จะเป็นการป้อนข้อมูลและตรวจสอบรายละเอียดต่างๆ ของเกรดนักศึกษาแต่ละคน แต่ละวิชา โดยจะเป็นผลการศึกษาของนักศึกษาแต่ละคนที่ลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในเทอมนั้นๆ

- การแก้ไขเกรด เป็นป้อนข้อมูล แก้ไข ในกรณีที่บันทึกเกรดผิดพลาดหรือมีการเปลี่ยนแปลงเกรด หลังจากที่ได้ลงบันทึกไว้แล้ว
- สรุปผลเฉพาะภาคการศึกษา เป็นการตรวจสอบรายละเอียดต่างๆ ของภาคการศึกษานั้นๆ
- สรุปการศึกษาทั้งหมดที่ผ่านมา เป็นการตรวจสอบรายละเอียดต่างๆ ของการศึกษาทั้งหมดที่ผ่านมาของนักศึกษาแต่ละคนว่า แต่ละเทอมมีผลการศึกษาเป็นอย่างไร

3:2.2 ส่วนของฝ่ายอาจารย์

เป็นส่วนที่ อาจารย์เป็นผู้จัดกรงานต่างๆ เช่น แก้ไขเกรดของนักศึกษาของตนเอง หรือจะดูรายละเอียดต่างๆ ของแต่ละรายวิชาก็ได้, ตรวจสอบประวัติของนักศึกษา เป็นต้น โดยส่วนของฝ่ายอาจารย์จะมีระบบการทำงานดังนี้

1. เกรดของนักศึกษา มีรายละเอียดต่างๆ ดังนี้

- การบันทึก โดยจะป้อนเกรดของนักศึกษาแต่ละคนในแต่ละรายวิชาว่า นักศึกษาคนใดได้เกรดในรายวิชาที่ลงทะเบียนเรียนและมีอาจารย์เป็นผู้สอน
- การแก้ไข จะเป็นในกรณีที่อาจารย์คนนั้นต้องการจะแก้ เกรดของนักศึกษาแต่ละคนในแต่ละวิชา โดยจะต้องป้อน รหัสของนักศึกษา และป้อนเกรดที่เปลี่ยนใหม่ หลังจากนั้นต้องยืนยันการแก้ไขเกรดไปยังฝ่ายทะเบียนด้วย
- ตรวจสอบความถูกต้อง จะเป็นการตรวจสอบข้อมูลที่ป้อนเข้าไปว่าข้อมูลดังกล่าวถูกต้องหรือไม่

2. ตรวจสอบผลการลงทะเบียน มีรายละเอียดต่างๆ ดังนี้

- รายวิชาที่ต้องการทราบข้อมูล เป็นส่วนที่อาจารย์ว่า แต่ละวิชาดังกล่าว มีนักศึกษาลงทะเบียนกี่คน ใครบ้าง
- นักศึกษาที่ต้องการทราบข้อมูล เป็นส่วนที่อาจารย์ จะตรวจสอบว่านักศึกษาคนนั้นลงทะเบียนอะไรบ้าง

3. ข้อมูลของนักศึกษา เป็นส่วนที่เกี่ยวข้องกับนักศึกษาโดยตรง ซึ่งอาจารย์จะสามารถตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สอบ ประวัติของนักศึกษาได้ ซึ่งมีการทำงานเป็นดังนี้

- ประวัติส่วนตัวของนักศึกษา ในส่วนนี้ จะเป็นส่วนที่อาจารย์ผู้สอนต้องการทราบรายละเอียดของนักศึกษา ซึ่งจะสามารถค้นหาประวัตินักศึกษาที่ต้องการทราบได้โดยการป้อนรหัสนักศึกษาเข้าไป
- ประวัติการศึกษา เป็นส่วนที่ให้อาจารย์ที่ต้องการทราบรายละเอียด ประวัติการศึกษาของนักศึกษา ซึ่งก็จะสามารถค้นหานักศึกษาที่ต้องการทราบ วิธีการจะเป็นเช่นเดียวกับประวัติส่วนตัวของนักศึกษา

3.2.3 ส่วนของนักศึกษา

ในส่วนนี้ จะเป็นส่วนของนักศึกษาซึ่งนักศึกษาจะมีสิทธิในการตรวจสอบหรือแก้ไขในส่วนของนักศึกษาเองเท่านั้น และไม่มีสิทธิในการป้อนข้อมูลหรือแก้ไขส่วนอื่นๆ ได้ เช่น ส่วนของรายวิชาที่เปิดสอน หรือ ประวัติส่วนตัวของนักศึกษา แต่สามารถป้อนข้อมูลรายละเอียดต่างๆ ได้เฉพาะในส่วนของการลงทะเบียนเรียนเท่านั้น โดยนักศึกษาจะเป็นคนที่ทำการป้อนข้อมูลเพิ่มเติมหรือแก้ไขรายวิชาที่ต้องลงทะเบียนซึ่งจะเป็นส่วนที่กำหนดมาจากฝ่ายทะเบียน ในส่วนของนักศึกษานี้ จะมีการทำงานเป็นดังนี้

1. ประวัติส่วนตัว นักศึกษาสามารถจะตรวจสอบดูประวัติของตัวเองได้ว่าถูกต้องหรือไม่ ซึ่งถ้าถูกต้องจะไม่มีสิทธิในการแก้ไข จะสามารถทำการแก้ไขโดยการยื่นหนังสือไปยัง ฝ่ายทะเบียน ให้ฝ่ายทะเบียนเป็นผู้แก้ไข
2. ผลการศึกษา
 - ตรวจสอบผลการศึกษาเป็นรายวิชา ซึ่งสามารถตรวจสอบได้ว่า รายวิชาที่เรียนผ่านมาแล้วได้เกรดอะไร เพื่อประโยชน์ในการลงทะเบียนเรียน ทำได้โดยการป้อนรหัสวิชานั้นๆ ลงไป
 - ตรวจสอบเป็นปีการศึกษา
 - สรุปผลการศึกษา

3. การลงทะเบียน

- การลงทะเบียน ทำได้โดยการป้อนรหัสนักศึกษาลงไป ตลอดจนชั้นปี ภาคเรียนที่ เสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วจะขึ้นรายวิชาที่ต้องลงทะเบียนซึ่งจะกำหนดโดยฝ่ายทะเบียน นักศึกษาสามารถแก้ไขและเพิ่มเติมรายวิชาได้ โดยการป้อนรหัสวิชาดังกล่าว

4. รายละเอียดของรายวิชา เป็นส่วนที่เก็บข้อมูลรายละเอียดของวิชาต่างๆ ว่าเทอมนี้เปิดวิชาอะไรบ้าง อาจารย์ผู้สอน ช่วงเวลาเรียนและสอบ นักศึกษาจะสามารถทราบรายละเอียดได้จากส่วนนี้แต่จะไม่สามารถแก้ไขได้ จะทำได้เพียงแค่ตรวจสอบและดูรายละเอียดเพื่อประโยชน์ในการลงทะเบียนเรียนเท่านั้น โดยจะมีรายละเอียดดังนี้
- รายละเอียดต่างๆ ของวิชาบังคับเฉพาะภาคการศึกษาโดยจะมีรายละเอียดว่าในแต่ละภาคการศึกษา จะมีวิชาบังคับอะไรบ้างของภาควิชาอะไรบ้างที่ เปิดสอน
 - รายละเอียดต่างๆ ของวิชาเลือกเฉพาะภาคการศึกษาโดยจะมีรายละเอียดว่าในแต่ละภาคการศึกษา จะมีวิชาเลือกอะไรบ้างของภาควิชาอะไรบ้างที่ เปิดสอน
 - รายละเอียดของวิชาต่างๆ ที่เปิดสอนทั้งหมดในเทอมปัจจุบันโดยจะมีรายละเอียดเกี่ยวกับ รหัสวิชา, ชื่ออาจารย์ผู้สอน, ช่วงเวลาการเรียนและสอบ ในเทอมนั้น
 - รายละเอียดของวิชาต่างๆ ที่มีทั้งหมดในหลักสูตร โดยมีรายละเอียดเกี่ยวกับ รหัสวิชา ชื่อวิชา, จำนวนหน่วยกิตที่เป็นทฤษฎีและปฏิบัติ และวิชาที่ต้องลงเรียนและผ่านก่อนหน้า วิชานี้ เป็นต้น รายวิชาเหล่านี้อาจจะเปิดสอนในเทอมปัจจุบันก็ได้

3.3 การออกแบบส่วนนำข้อมูล

ในส่วนการนำข้อมูลเข้าสู่ระบบ จะเป็นส่วนที่ให้ผู้ใช้งานติดต่อกับระบบ ซึ่งจะแสดงในภาษาไทย ขนาด 25 บรรทัดได้ในขณะที่สามารถป้อนข้อมูลเป็นภาษาอังกฤษได้ด้วย ข้อความที่จะช่วยให้ผู้ใช้งานสามารถใช้งานระบบได้ จะเป็นภาษาไทย เพื่อให้เข้าใจได้ง่าย ในระบบทะเบียนนักศึกษาที่ออกแบบนี้ จะแบ่งการทำงานต่างๆ นี้เป็น 6 ส่วน คือ

- "ก่อนหน้า" หมายถึง ดูข้อมูลก่อนหน้าที่แสดงอยู่
- "หลัง" หมายถึง ดูข้อมูลหลังจากหน้าที่แสดงอยู่
- "เพิ่มเติม" หมายถึง ต้องการเพิ่มเติมข้อมูลเข้าไปใหม่
- "ค้นหา" หมายถึง ต้องการค้นหาข้อมูล
- "ลบ" หมายถึง ต้องการลบข้อมูล ซึ่งจะมีการให้ยืนยันการลบข้อมูล

ถ้าเป็นกรณีนอกเหนือจากนี้ จะมีการอธิบายให้เข้าใจได้ ถ้าเกิดข้อผิดพลาดขึ้นจะ error message เป็นภาษาไทยให้

3.4 โครงสร้างแฟ้มข้อมูล

1. โครงสร้างแฟ้มข้อมูลนักศึกษา

ชื่อแฟ้มข้อมูล : STU.BTR

ลักษณะแฟ้มข้อมูล : Master File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTREVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	stu_id	character	10	รหัสนักศึกษา
2.	part	integer	2	ภาควิชา
3.	th_name	character	21	ชื่อภาษาไทย
4.	sex	integer	2	เพศ
5.	th_sname	character	31	นามสกุลภาษาไทย
6.	cname	character	21	ชื่อภาษาอังกฤษ
7.	e_sname	character	31	นามสกุลภาษาอังกฤษ
8.	d_birth	character	9	วันเกิด
9.	c_cont	character	9	วันเข้ารับการศึกษ
10.	adv_name	character	41	ชื่อ อ. ที่ปรึกษา
11.	add1	character	41	ที่อยู่ปัจจุบัน 1
12.	add2	character	41	ที่อยู่ปัจจุบัน 2
13.	tel	character	9	เบอร์โทรศัพท์
14.	p_name	character	21	ชื่อผู้ปกครอง
15.	p_sname	character	31	นามสกุลผู้ปกครอง
16.	p_add1	character	41	ที่อยู่ผู้ปกครอง 1
17.	p_add2	character	41	ที่อยู่ผู้ปกครอง 2
18.	p_tel	character	9	เบอร์โทรศัพท์ผู้ปกครอง
19.	stu_cred	integer	2	จำนวนหน่วยกิตรวม
20.	stu_sco	character	4	คะแนนรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. โครงสร้างแฟ้มข้อมูลประวัติการศึกษา

ชื่อแฟ้มข้อมูล : RESUME.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	stu_id	character	10	รหัสนักศึกษา
2.	locate1	character	41	ชื่อร.ร ประถมศึกษา
3.	time_int1	character	10	ช่วงเวลาที่เรียน
4.	locate2	character	41	ชื่อร.ร มัธยม ต้น
5.	time_int2	character	10	ช่วงเวลาที่เรียน
6.	locate3	character	41	ชื่อร.ร มัธยม ปลาย
7.	time_int3	character	10	ช่วงเวลาที่เรียน
8.	locate4	character	41	ชื่อสถานศึกษาอุดมศึกษา
9.	time_int4	character	10	ช่วงเวลาที่เรียน
	รวม		214	

2. โครงสร้างแฟ้มข้อมูลโปรแกรมวิชาการศึกษา

ชื่อแฟ้มข้อมูล : RESUME.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	stu_id	charactor	10	รหัสนักศึกษา
2.	locate1	charactor	41	ชื่อร.ร ประถมศึกษา
3.	time_int1	charactor	10	ช่วงเวลาที่เรียน
4.	locate2	charactor	41	ชื่อร.ร มัธยม ต้น
5.	time_int2	charactor	10	ช่วงเวลาที่เรียน
6.	locate3	charactor	41	ชื่อร.ร มัธยม ปลาย
7.	time_int3	charactor	10	ช่วงเวลาที่เรียน
8.	locate4	charactor	41	ชื่อสถานศึกษาอุดมศึกษา
9.	time_int4	charactor	10	ช่วงเวลาที่เรียน
	รวม		214	

3. โครงสร้างแฟ้มข้อมูลรายวิชาทั้งหมด

ชื่อแฟ้มข้อมูล : TSUB.BTR

ลักษณะแฟ้มข้อมูล : Master File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTREIVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	subj_code	character	7	รหัสวิชา
2.	subj_name	character	61	ชื่อวิชาภาษาไทย
3.	subj_engname	character	61	ชื่อวิชาภาษาอังกฤษ
4.	subj_grad	integer	2	หน่วยกิต
5.	subj_theory	integer	2	ทฤษฎี
6.	subj_opr	integer	2	ปฏิบัติ
7.	subj_prev1	character	7	รหัสวิชาก่อนหน้า 1
8.	subj_prev2	character	7	รหัสวิชาก่อนหน้า 2
9.	subj_prev3	character	7	รหัสวิชาก่อนหน้า 3
รวม			156	

4. โครงสร้างแฟ้มข้อมูลรายวิชาที่เปิดสอน

- ชื่อแฟ้มข้อมูล : OSUB.BTH
ลักษณะแฟ้มข้อมูล : Transaction File
โครงสร้าง : Balance B-Tree
โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	osub_id	charactor	7	รหัสวิชาที่เปิดสอน
2.	osub_tea	charactor	31	ชื่ออาจารย์ผู้สอน
3.	osub_day	integer	2	วันที่ทำการเรียน,สอน
4.	osub_part1	integer	2	ช่วงเวลาที่เรียน,สอน
5.	osub_date	charactor	7	วันสอบ
6.	osub_part2	integer	2	ช่วงเวลาที่ทำการสอบ
	รวม		51	

5. โครงสร้างแฟ้มข้อมูลโครงสร้างหลักสูตร

ชื่อแฟ้มข้อมูล : PLAN.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTREIVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	plan_dep	integer	2	ภาควิชา
2.	plan_sem	integer	2	ภาคการศึกษา
3.	plan_year	integer	2	ชั้นปีการศึกษา
รายละเอียด				
1.	code	charactor	7	รหัสวิชา 1
	รวม		13	

6. โครงสร้างแฟ้มข้อมูลการลงทะเบียน

- ชื่อแฟ้มข้อมูล : REGIS.BTR
- ลักษณะแฟ้มข้อมูล : Transaction File
- โครงสร้าง : Balance B-Tree
- โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว (bytes)	ความหมาย
1.	id	charactor	10	รหัสนักศึกษา
2.	depp	integer	2	ภาควิชา
3.	sem	integer	2	ภาคการศึกษา
4.	year	integer	2	ชั้นปีการศึกษา
รายละเอียด				
1.	code	charactor	7	รหัสวิชา

7. โครงสร้างแฟ้มข้อมูลผลการศึกษา

ชื่อแฟ้มข้อมูล : GRADE.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	code	charactor	10	รหัสนักศึกษา
2.	depp	integer	2	ภาควิชา
3.	sem	integer	2	ภาคการศึกษา
4.	year	integer	2	ชั้นปีการศึกษา
รายละเอียด				
1.	code	charactor	7	รหัสวิชา
2.	grd	charactor	3	ผลการศึกษา

9. โครงสร้างแฟ้มข้อมูลวิชาเลือก

ชื่อแฟ้มข้อมูล : SELECT.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	selct_dep	integer	2	ภาควิชา
2.	selct_year	integer	2	ภาคการศึกษา
3.	ksub	integer	2	ประเภทของวิชาเลือก
4.	slcode	charactor	7	รหัสวิชา
รวม			13	

10. โครงสร้างแฟ้มข้อมูลการกำหนดค่าการลงทะเบียน

ชื่อแฟ้มข้อมูล : INTER.BTR
ลักษณะแฟ้มข้อมูล : Transaction File
โครงสร้าง : Balance B-Tree
โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	st1	charactor	9	เริ่มต้นการลงทะเบียน
2.	end1	charactor	9	สิ้นสุดการลงทะเบียน
3.	st2	charactor	9	เริ่มต้นการเพิ่มถอนวิชา
4.	end2	charactor	9	สิ้นสุดการเพิ่มถอนวิชา
รวม			36	

11. โครงสร้างแฟ้มข้อมูลการเปลี่ยนหรือแก้ไขเกรด

ชื่อแฟ้มข้อมูล : CGR.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	cgr_sub	character	7	รหัสวิชา
2.	code	character	10	รหัสนักศึกษา
3.	grade	character	3	เกรดปัจจุบัน

รวม

20

12. โครงสร้างแฟ้มข้อมูลการแก้ไขเกรดของอาจารย์

ชื่อแฟ้มข้อมูล : TGR.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	tgr_sub	character	7	รหัสวิชา
2.	code	character	10	รหัสนักศึกษา
3.	grade1	character	3	เกรดเก่า
4.	grade2	character	3	เกรดที่แก้ไข
รวม			23	

13. โครงสร้างแฟ้มข้อมูลสรุปผลการลงทะเบียน

ชื่อแฟ้มข้อมูล : CSEM.BTR

ลักษณะแฟ้มข้อมูล : Transaction File

โครงสร้าง : Balance B-Tree

โปรแกรมจัดการ : BTRIEVE RECORD MANAGER

ลำดับ	ชื่อตัวแปร	รูปแบบ	ความยาว(bytes)	ความหมาย
1.	code	charactor	10	รหัสนักศึกษา
2.	dep	integer	2	ภาควิชา
3.	sem	integer	2	ภาคการศึกษา
4.	year	integer	2	ชั้นปีที่
5.	oldtot	integer	2	หน่วยกิตสะสม
6.	total	integer	2	หน่วยกิตสะสมใหม่
7.	gps	float	4	เกรดเทอม
8.	gta	float	4	เกรดเฉลี่ย
9.	sco	float	4	คะแนนรวม
10.	status	charactor	10	สถานภาพนักศึกษา
รวม			52	

3.5 การออกแบบหน้าจอ

คณะวิทยาศาสตร์
ระบบทะเบียนนักศึกษา (ฝ่ายทะเบียน)
1. ประวัตินักศึกษา 2. การบันทึกรายวิชา 3. การลงทะเบียนนักศึกษา 4. ผลการศึกษาของนักศึกษา 9. ออกจากระบบ

กด F1- F4 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 3.1 แสดงหน้าจอเมนูหลักของฝ่ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์

ประวัตินักศึกษา

1. การบันทึกประวัติส่วนตัว
2. การบันทึกประวัติการศึกษา
9. ออกสู่มายการหลัก

กด F1- F2 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

คณะวิทยาศาสตร์
การบันทึกรายวิชา
1. หมวดวิชาบังคับเฉพาะภาคการศึกษา 2. หมวดวิชาเลือกเฉพาะภาคการศึกษา 3. รายละเอียดวิชาที่เปิดสอนในปัจจุบัน 4. รายละเอียดวิชาทั้งหมด 5. โครงสร้างหลักสูตร 9. ออกสู่รายการหลัก

กต F1- F5 หรือ ลูกศรขึ้น-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 3.3 แสดงหน้าจอเมนูของการบันทึกรายวิชา

คณะวิทยาศาสตร์

การลงทะเบียนนักศึกษา

1. กำหนดช่วงวันทะเบียน
2. สรุปผลการลงทะเบียน
9. ออกสูัรรายการหลัก

กค F1- F2 หรือ ลูกศรขึ้น-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 3.4 แสดงหน้าจอเมนูของการลงทะเบียนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
ผลการเรียนของนักศึกษา 1. การบันทึกเกรด 2. การแก้ไขเกรด 3. สรุปผลเฉพาะภาคการศึกษา 4. สรุปผลทั้งหมดที่ผ่านมา 9. ออกสู่อายการหลัก

กด F1- F4 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 3.5 แสดงหน้าจอเมนูผลการเรียนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา: 99-999999

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อ(ไทย): XXXXXXXXXXXXXXXXXXXX

เพศ: XXX

นามสกุล(ไทย): XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME: XXXXXXXXXXXXXXXXXXXX

SURNAME: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด: 99/99/99

[dd/mm/yy]

วันเข้ารับการศึกษ: 99/99/99

[dd/mm/yy]

อ.ที่ปรึกษา: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

ชื่อผู้ปกครอง: XXXXXXXXXXXXXXXXXXXX

นามสกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

หน่วยกิตสะสม: 999

1:ก่อน 2:หลัง 3:ค้นหา 4:แก้ไข 5:เพิ่มเติม 6:ลบ 9:Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.6 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลประวัตินักศึกษา		
รหัสนักศึกษา: 99-999999		
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
ภาควิชา: XXXXXXXXXXXXXXXXXXXXXXXX		
ระดับการศึกษา	สถานศึกษา	ช่วงปีการศึกษา
ประถมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXXX	9999-9999
ประถมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXXX	9999-9999

1:ก่อน 2:หลัง 3:ค้นหา 4:แก้ไข 5:เพิ่มเติม 6:ลบ 9:Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.7 แสดงหน้าจอของประวัติการศึกษาของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวดวิชาบังคับ	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	
ชั้นปีที่: 9	ภาคเรียนที่: 9
รหัสวิชา: 99-999	
ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX	
หน่วยกิต: 9	
ทฤษฎี: 9	ปฏิบัติ: 9

1:ก่อน 2:หลัง 3:แก้ไข 4:เพิ่มเติม 5:ลบ 9:Exit
กด F1 - F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.8 แสดงหน้าจอของรายละเอียดวิชาบังคับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาเลือกเฉพาะภาคการศึกษา	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	ภาคเรียนที่: 9
วิชาเลือก: XXXXXXXXXXXXXXXXXXXX รหัสวิชา: 99-999 ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX หน่วยกิต: 9	

1:ก่อน 2:หลัง 3:แก้ไข 4:เพิ่มเติม 5:ลบ 9:Exit

กด F1 – F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.9 แสดงหน้าจอของรายละเอียดวิชาเลือกเฉพาะภาคการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลกระบวนวิชาที่เปิดสอนในเทอมปัจจุบัน

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา (อังกฤษ): XXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันสอน: 9 เวลาเรียน: 9

วันสอบ: 99/99 [dd/mm] ช่วง: 9

1:ก่อน 2:หลัง 3:ค้นหา 4:แก้ไข 5:เพิ่มเติม 6:ลบ 9:Exit
กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.10 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนในเทอมนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายที่วิชาทั้งหมดในหลักสูตร

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา (อังกฤษ): XXXXXXXXXXXXXXXXXXXX

หน่วยกิต: 9

ทฤษฎี: 9

ปฏิบัติ: 9

วิชาก่อนหน้า: 99-999 99-999 99-999

1: ก่อน 2: หลัง 3: ค้นหา 4: แก้ไข 5: เพิ่มเติม 6: ลบ 9: Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.11 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนทั้งหมดในหลักสูตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น. ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างการศึกษา			
ภาควิชา: XXXXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
รหัสวิชา	ชื่อวิชา	หน่วยกิต	ทฤษฎี-ปฏิบัติ
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9

1:ก่อน 2:หลัง 3:แก้ไข 4:เพิ่มเติม 5:ลบ 9:Exit
กด F1 - F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.12 แสดงหน้าจอของโครงสร้างหลักสูตรทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดช่วงวันลงทะเบียน	
การลงทะเบียนปกติ	
เริ่มต้น: 99/99/99	[dd/mm/yy]
สิ้นสุด: 99/99/99	[dd/mm/yy]
กรณีเพิ่มหรือถอน	
เริ่มต้น: 99/99/99	[dd/mm/yy]
สิ้นสุด: 99/99/99	[dd/mm/yy]

1:หลัง 2:แก้ไข 3:เพิ่มเติม 9:Exit
 กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.13 แสดงหน้าจอของการกำหนดช่วงของการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลการลงทะเบียน

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXX

ภาคเรียนที่: 9 ชั้นปีที่: 9

รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXXX	9

รวม : 999

1:เพิ่มเติม 2:ลบ 3:นักศึกษารับ 9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.14 แสดงหน้าจอของการลงทะเบียนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลผลการศึกษาประจำภาคการศึกษา

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXXXX

ภาคเรียนที่: 9 ชั้นปีที่: 9

รหัสวิชา	ชื่อวิชา	หน่วยกิต	ผลการศึกษา
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX

รวม : 999 เดลีย : 9.999

1:ผลกรรศึกษนักรรศึกษอื่น 9:Exit

Esc-ขกรเลกร

F10-ดกรลง

รูปที่ 3.15 แสดงหน้าจอรเมื่อเรตต้องการป้อนผลกรรศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแก้ไขเฉพาะรายวิชา	
รหัสวิชา: 99-999	
ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX	
อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	

รหัสนักศึกษา: 99-999999	
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	
เกรด: XX	
เกรดที่แก้ไข: XX	
เกรดปัจจุบัน: XX	

1:ก่อน 2:หลัง 3:ค้นหา 4:แก้ไข 5:เพิ่มเติม 6:ลบ 9:Exit

กด F1 - F6 หรือ ลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.16 แสดงหน้าจอของการแก้ไขเฉพาะรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการศึกษาเฉพาะภาคเรียน

รหัสนักศึกษา: 99-999999

ชื่อ-สกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXXXX

ภาคเรียนที่: 9

ชั้นปีที่: 9

	เฉพาะเทอมนี้	ของเดิม	ของใหม่
จำนวนหน่วยกิต	999	999	999
เกรดที่ได้ (GPS/GPA)	9.999	9.999	9.999

1:ก่อน 2:หลัง 3 :นักศึกษาอื่น 9:Exit
กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.17 แสดงหน้าจอของการสรุปผลการศึกษาเฉพาะภาคเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนวิชาศาสตร์
ระบบทะเบียนนักศึกษา (อาจารย์)
1. เกรดของนักศึกษา 2. ผลการลงทะเบียนเฉพาะราย 3. ข้อมูลของนักศึกษา 9. ออกจากระบบ

กต F1 - F3 หรือ ลูกศรบน-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 3.18 แสดงรายการงานของส่วนของอาจารย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์

เกรดของนักศึกษา

1. การบันทึกเกรดเฉพาะรายวิชา
2. ผลการศึกษานักศึกษาเฉพาะราย
9. ออกสู่อายการหลัก

กต F1 - F2 หรือ ลูกศรบน-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 3.19 แสดงงานในส่วนของเกรดนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกเกรดเฉพาะรายวิชา

รหัสวิชา : XX-XXXX

ชื่อวิชา : XXXXXXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน : XXXXXXXXXXXXXXXXXXXXXXXX

รหัสนักศึกษา : 99-999999

ชื่อ : XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา : XXXXXXXXXXXXXXXXXXXXXXXX

เกรด : XXX

เกรดแก้ไข : XXX

1:ก่อน

2:หลัง

3:ค้นหา

4:แก้ไข

5:เพิ่มเติม

6:ลบ

9:Exit

กด F1 - F6 หรือ ลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.20 แสดงการทำงานในส่วนของการเปลี่ยนแปลงแก้ไขผลการศึกษาเป็นรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการศึกษานักศึกษาเฉพาะราย

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ภาคเรียนที่: 9 ชั้นปีที่: 9

รหัสวิชา	ชื่อวิชา	หน่วยกิต	ผลการศึกษา
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX

รวม : 999 เฉลี่ย : 9.999

1: ผลการศึกษานักศึกษาอื่น 9: Exit

Esc-ยกเลิก

F10-ตกลง

รูปที่ 3.21 แสดงหน้าของการเปลี่ยนแปลงแก้ไขผลการศึกษาของนักศึกษา

คณะวิทยาศาสตร์
ข้อมูลนักศึกษา 1. ประวัติส่วนตัว. 2. ประวัติการศึกษา 9. ออกสู่มายการหลัก

กต F1 - F2 หรือ ลูกศรบน-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 3.22 แสดงหน้าของการตรวจสอบข้อมูลของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา : 99-999999

ภาควิชา : XXXXXXXXXXXXXXXXXXXX

ชื่อ (ไทย) : XXXXXXXXXXXXXXXXXXXX

เพศ : XXX

นามสกุล (ไทย) : XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME : XXXXXXXXXXXXXXXXXXXX

SURNAME : XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด : 99/99/99 [dd/mm/yy]

วันเข้ารับการศึกษ : 99/99/99 [dd/mm/yy]

อ.ที่ปรึกษา : XXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน : XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์ : 999-9999

ชื่อผู้ปกครอง : XXXXXXXXXXXXXXXXXXXX

นามสกุล : XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน : XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์ : 999-9999

หน่วยกิตสะสม : 999

1:ก่อน 2:หลัง 3:ค้นหา

9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.23 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลประวัตินักศึกษา

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ระดับการศึกษา	สถานศึกษา	ช่วงปีการศึกษา
ประถมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
ประถมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999

1:ก่อน 2:หลัง 3:ค้นหา

9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

ข้อมูลการลงทะเบียนของนักศึกษาเฉพาะราย		
รหัสนักศึกษา: 99-999999		
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
ภาควิชา: XXXXXXXXXXXXXXXX		
ภาคเรียนที่: 9 ชั้นปีที่: 9		
รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9

รวม : 999

1:นักศึกษาอื่น

9:Exit

กด F1 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.25 แสดงหน้าจอตรวจสอบการลงทะเบียนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์

ระบบทะเบียนนักศึกษา (นักศึกษา)

1. ประวัติส่วนตัว
2. ตรวจสอบผลการศึกษา
3. ลงทะเบียนและเพิ่ม, ถอนวิชาเรียน
4. ตรวจสอบรายละเอียดวิชาที่เปิดสอน
9. ออกจากระบบ

กด F1 - F4 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 3.26 แสดงรายการงานของส่วนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้อนรหัสนักศึกษา

รหัสนักศึกษา : XX-XXXXXX

ESC-ชกเล็ก

F10-ตกลง



รูปที่ 3.27 แสดงหน้าจอให้ป้อนรหัสนักศึกษาเพื่อให้เห็นประวัติของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา: 99-999999

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อ (ไทย): XXXXXXXXXXXXXXXXXXXX

เพศ: XXX

นามสกุล (ไทย): XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME: XXXXXXXXXXXXXXXXXXXX

SURNAME: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด: 99/99/99 [dd/mm/yy]

วันเข้ารับการศึกษ: 99/99/99 [dd/mm/yy]

อ.ที่ปรึกษา: XXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

ชื่อผู้ปกครอง: XXXXXXXXXXXXXXXXXXXX

นามสกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

หมายเลขสะสม: 999

9:Exit

กค F1 - F1 หรือลูกสรขึ้น-ลง และกค RETURN

รูปที่ 3.28 แสดงประวัติของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์

ตรวจสอบผลการศึกษา

1. ตรวจสอบเป็นรายวิชา
2. ตรวจสอบผลการศึกษาเฉพาะภาคเรียน
9. ออกสู่วางการหลัก

กต F1 - F2 หรือ ลูกศรขึ้น-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 3.29 แสดงรายการย่อยของการตรวจสอบผลการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบผลการศึกษารายวิชา

รหัสวิชา : XX-XXX

ชื่อวิชา : XXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ) : XXXXXXXXXXXXXXXXXX

หน่วยกิต : 9

วันที่ลงทะเบียน : 99/99/99

เกรด : XX

Search Window

รหัสวิชา: XX-XXX

1: ต้องการตรวจสอบวิชาอื่น

9: ไม่ต้องการ หรือ Exit

ESC-ยกเลิก

F10-ตกลง

กรุณาใส่รหัสวิชา

รูปที่ 3.30 แสดงการตรวจสอบผลการศึกษาเป็นรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการศึกษาเฉพาะภาคเรียน

รหัสนักศึกษา : 99-999999

ชื่อ-สกุล : XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา : XXXXXXXXXXXXXXXXXXXX

ภาคเรียนที่ : 9

ชั้นปีที่ : 9

	เฉพาะเทอมนี้	ของเดิม	ของใหม่
จำนวนหน่วยกิต	999	999	999
เกรดที่ได้(GPS/GPA)	9.999	9.999	9.999

1:นักศึกษาอื่น

9:Exit

กด F1 - F1 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.31 แสดงหน้าจอการตรวจสอบผลการศึกษาเป็นภาคการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลการลงทะเบียน

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXX

ภาคเรียนที่: 9 ชั้นปีที่: 9

รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
.....		.
รวม: _____		999

1:เพิ่มเติมวิชา 2:ถอนวิชา 3:นักศึกษาอื่น

9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 3.32 แสดงการลงทะเบียนเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

คณะวิทยาศาสตร์
รายละเอียดของรายวิชา
1. วิชาทั้งหมดในหลักสูตร 2. วิชาบังคับเฉพาะภาคการศึกษา 3. วิชาเลือกเฉพาะภาคการศึกษา 4. วิชาที่เปิดสอนในปัจจุบัน 9. ออกสัรรายการหลัก

กค F1 – F4 หรือ ลูกศรขึ้น-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 3.33 แสดงรายการย่อยของการตรวจสอบรายละเอียดวิชาเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาบังคับเฉพาะภาคการศึกษา

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ชั้นปีที่: 9

ภาคเรียนที่: 9

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

หน่วยกิต: 9

ทฤษฎี: 9

ปฏิบัติ: 9

1: ก่อน

2: หลัง

9: Exit

กค F1 - F2 หรือลูกศรขึ้น-ลง และกค RETURN

รูปที่ 3.34 แสดงรายละเอียดวิชาเรียนที่เป็นวิชาบังคับของภาคการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาเลือกเฉพาะภาคการศึกษา	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	ภาคเรียนที่: 9
วิชาเลือก: XXXXXXXXXXXXXXXXXXXX รหัสวิชา: 99-999 ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX หน่วยกิต: 9	

1: ก่อน

2: หลัง

9: Exit

กค F1 – F2 หรือลูกศรขึ้น-ลง และกค RETURN

รูปที่ 3.35 แสดงรายละเอียดวิชาเรียนที่เป็นวิชาเลือกของภาคการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายวิชาที่เปิดสอนในเทอมปัจจุบัน

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ): XXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันสอน: 9

เวลาเรียน: 9

วันสอบ: 99/99 [dd/mm]

ช่วง: 9

Search Window

รหัสวิชา: 99-999

1: ก่อน

2: หลัง

3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

ESC-ยกเลิก

F10-ตกลง

รูปที่ 3.36 แสดงรายละเอียดวิชาที่เปิดสอนในปัจจุบัน

รายละเอียดวิชาเปิดสอนในหลักสูตร

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา (อังกฤษ): XXXXXXXXXXXXXXXXXXXX

หน่วยกิต: 9

ทฤษฎี: 9

ปฏิบัติ: 9

วิชาก่อนหน้า: 99-999 99-999 99-999

Search Window

รหัสวิชา : 99-999

1: ก่อน 2: หลัง 3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

ESC-ยกเลิก

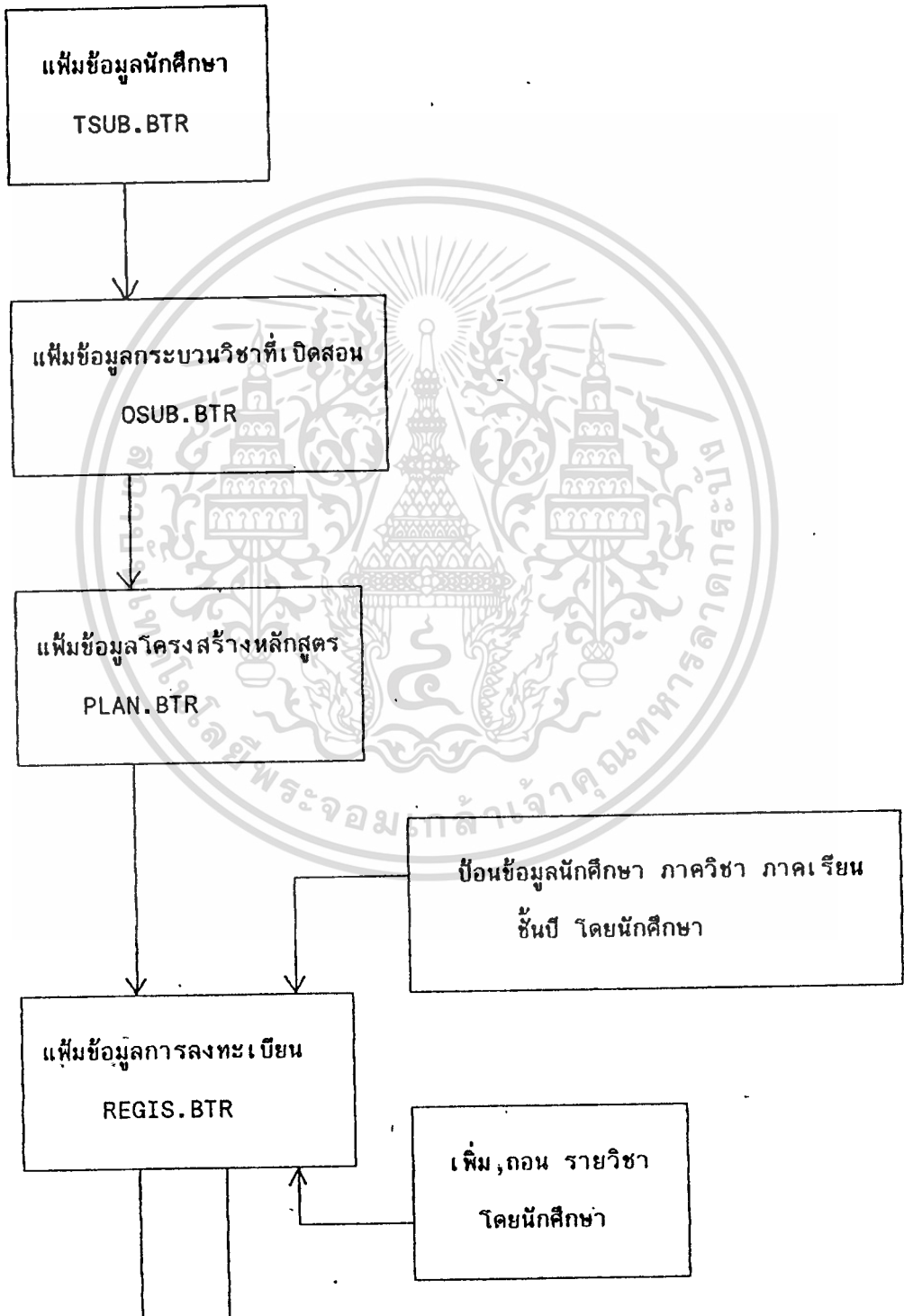
F10-ตกลง

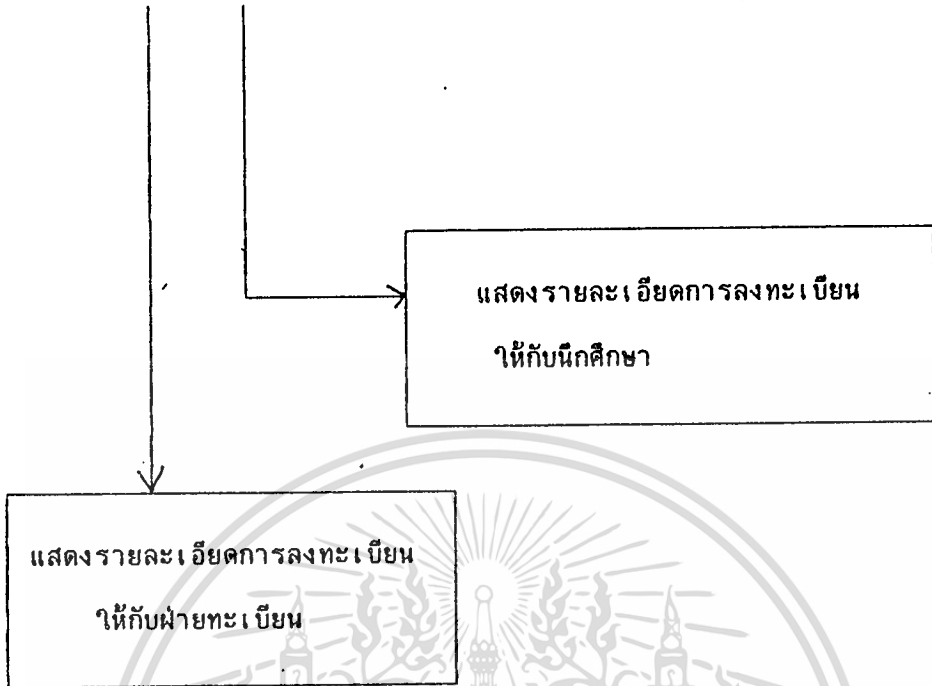
รูปที่ 3.37 แสดงรายละเอียดวิชาเรียนที่อยู่ในหลักสูตรทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

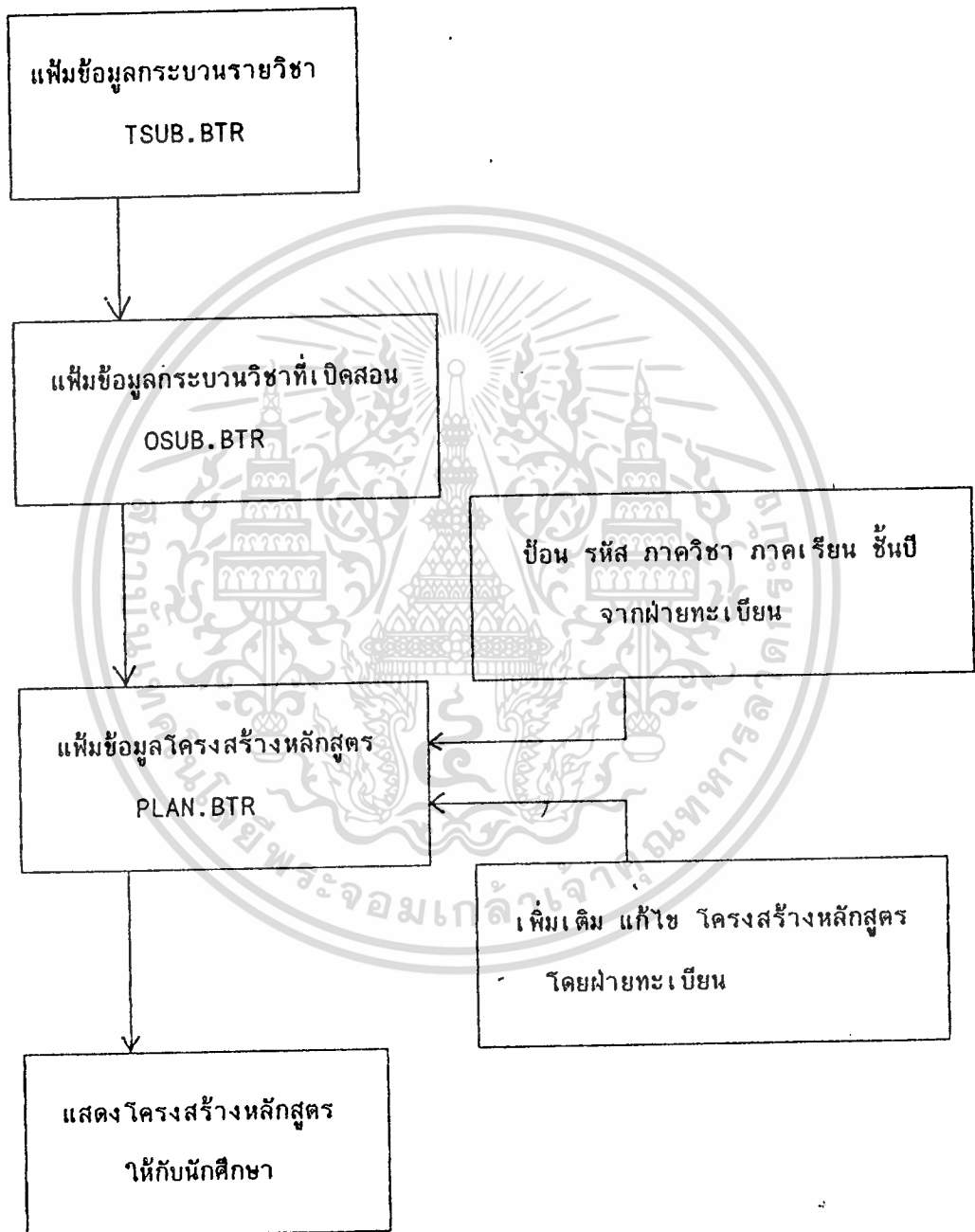
แผนผังแสดงการทำงานในระบบทะเบียนนักศึกษา

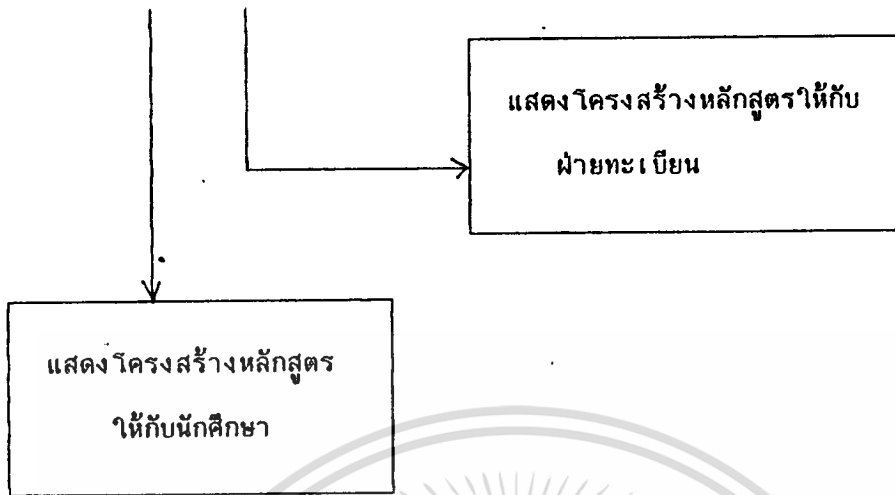
1. แผนผังแสดงระบบการลงทะเบียน



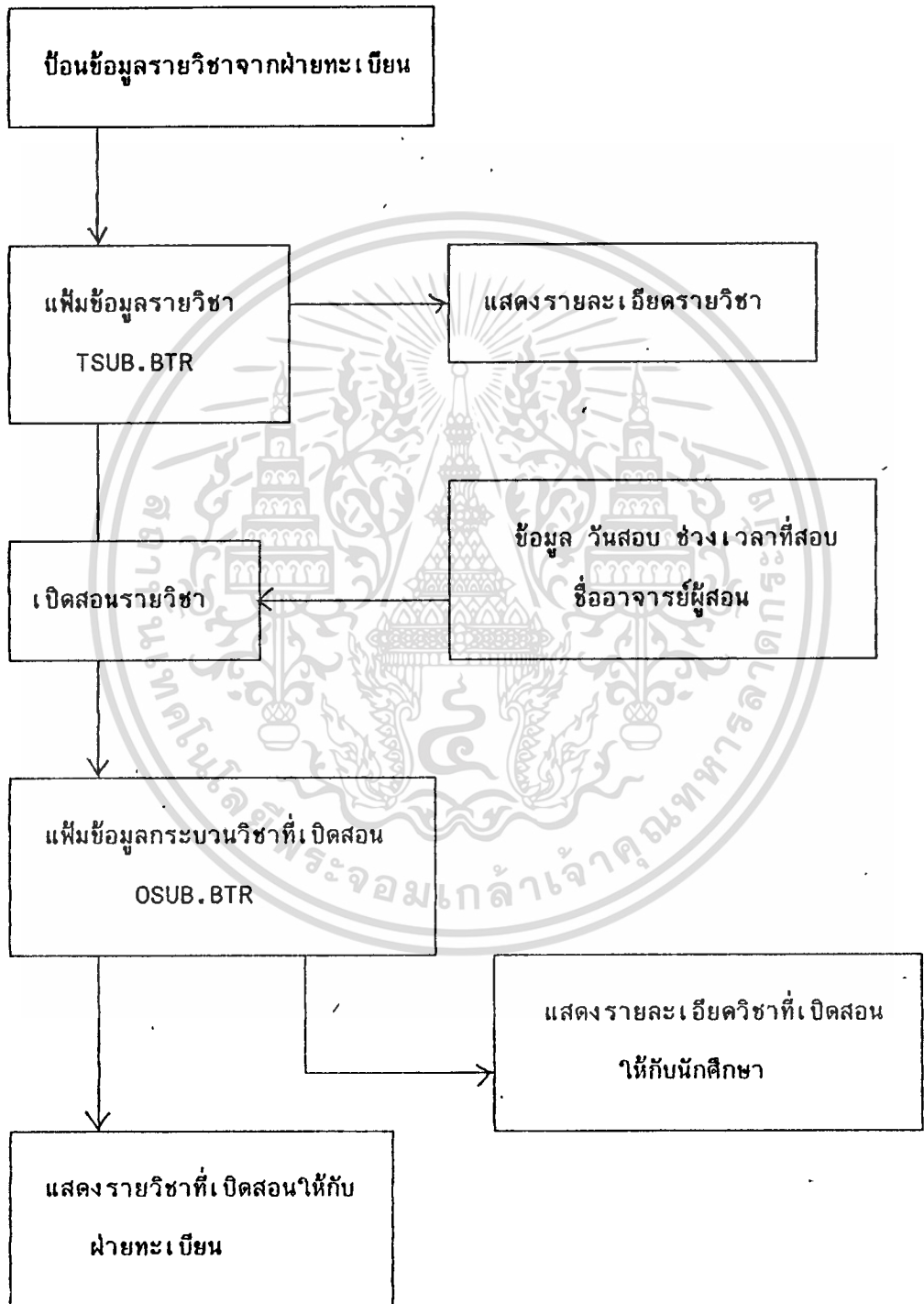


2. แผนผังแสดงระบบการออกแบบโครงสร้างข้อมูล

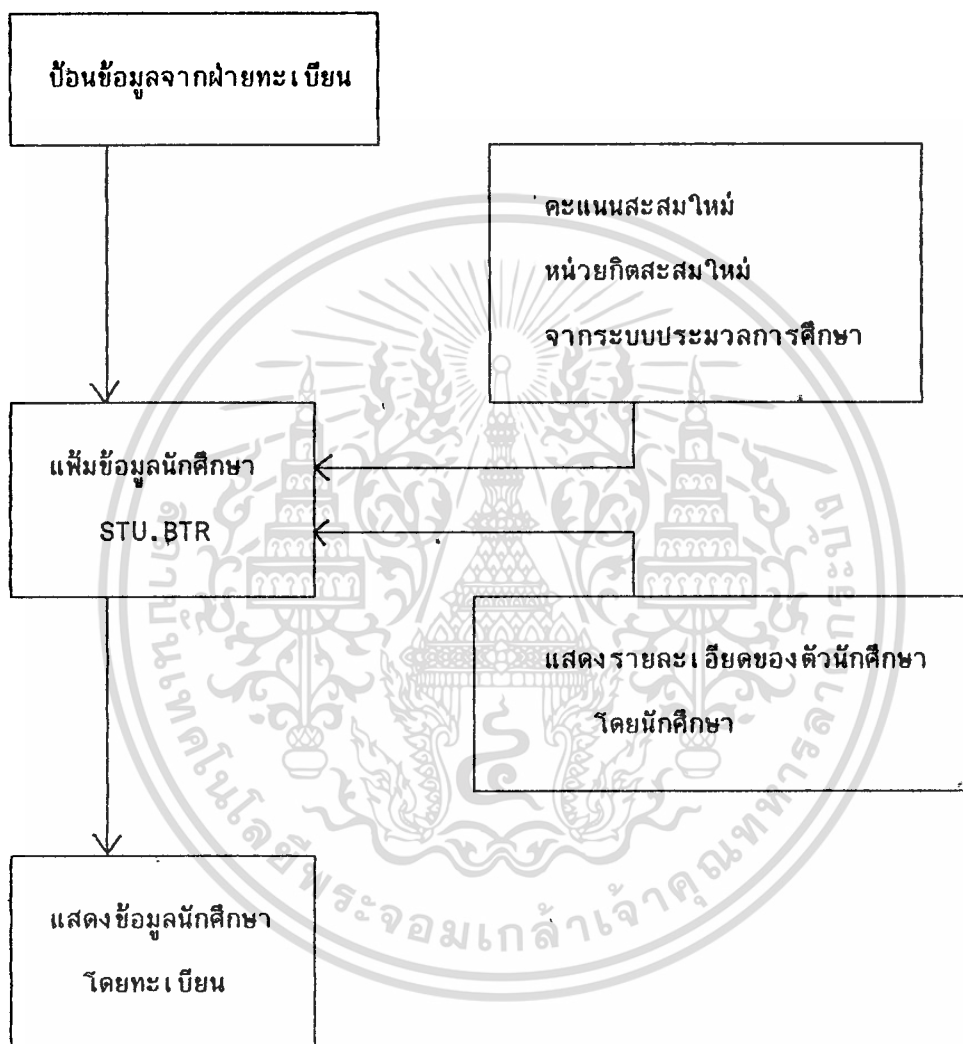


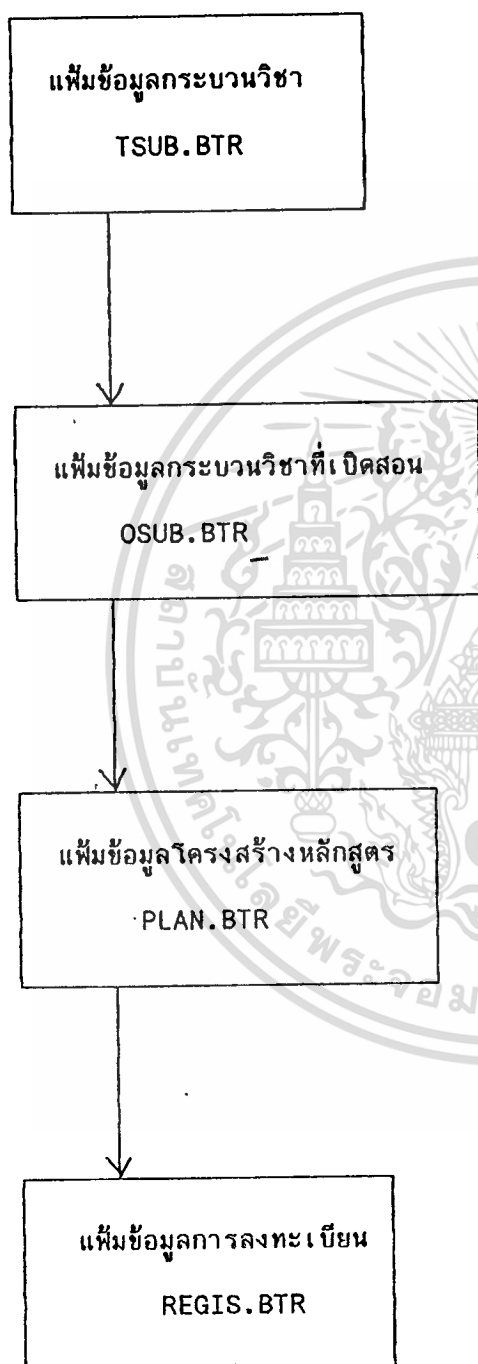


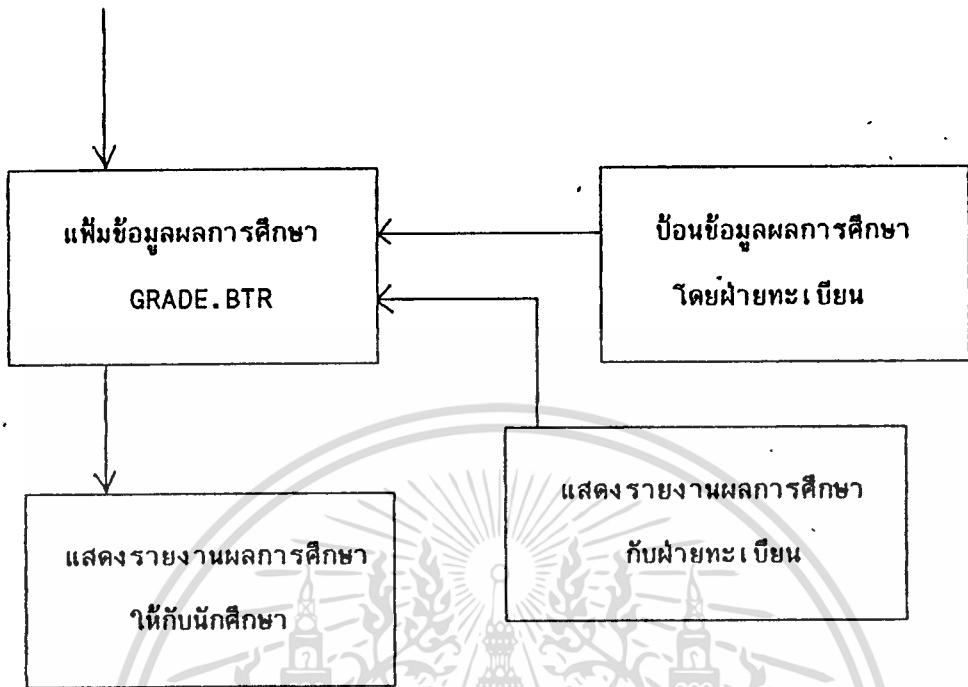
3. ผังงานแสดงระบบข้อมูลรายวิชาและระบบการเปิดสอน



4. ผังงานแสดงการทำงานของระบบข้อมูลนักศึกษา



5. แผนผังแสดงระบบป้อนข้อมูลผลการศึกษา



บทที่ 4 ขั้นตอนการใช้งานโปรแกรม

4.1 คู่มือการใช้งานระบบทะเบียนนักศึกษา

โปรแกรมระบบงานทะเบียนนักศึกษาจะมีโปรแกรมหลัก 3 โปรแกรม คือ

- P.EXE เป็นโปรแกรมระบบงานของฝ่ายทะเบียน
- TE.EXE เป็นโปรแกรมระบบงานของฝ่ายอาจารย์
- PRO.EXE เป็นโปรแกรมระบบงานของฝ่ายนักศึกษา

โดยมีโปรแกรมการจัดการฐานข้อมูลชื่อว่า BTRIEVE.EXE เป็นตัวจัดการข้อมูลต่าง ๆ การใช้งานโปรแกรมนี้ จะต้องทำงานบนระบบ PC LAN และที่ WORK STATION ทุกเครื่องควรจะมีระบบภาษาไทย 25 บรรทัดของบริษัท IRC จำกัด รุ่น IRC Standard

4.2 การใช้งานโปรแกรมระบบฝ่ายทะเบียน

สำหรับฝ่ายทะเบียน เมื่อเลือกใช้งานโปรแกรม P.EXE และป้อนรหัสผ่านเข้าสู่การทำงานเรียบร้อยแล้วจะเข้าสู่การเลือกรายการงานหลัก ตามรูปที่ 4.1 ซึ่งประกอบด้วย

1. ประวัตินักศึกษา
2. การบันทึกรายวิชา
3. การลงทะเบียนนักศึกษา
4. ผลการศึกษาของนักศึกษา

ประวัตินักศึกษา ประกอบด้วย

- การบันทึกประวัติส่วนตัว
- การบันทึกประวัติการศึกษา

การบันทึกรายวิชา ประกอบด้วย

- วิชาบังคับเฉพาะภาคการศึกษา
- วิชาเลือกเฉพาะภาคการศึกษา
- รายละเอียดวิชาที่เปิดสอนปัจจุบัน
- รายละเอียดรายวิชาทั้งหมด
- โครงสร้างหลักสูตร

การลงทะเบียนนักศึกษา ประกอบด้วย

- กำหนดช่วงวันลงทะเบียน
- สรุปผลการลงทะเบียน

การลงทะเบียนนักศึกษา ประกอบด้วย

- การบันทึกเกรด
- การแก้ไขเกรด
- สรุปผลเฉพาะภาคการศึกษา
- สรุปผลทั้งหมดที่ผ่านมา

ขั้นตอนการทำงาน

ประวัตินักศึกษา

การเตรียมข้อมูล

เนื่องจากระบบการทำงานทะเบียนนักศึกษา มีความเกี่ยวข้องกับประวัตินักศึกษา จึงต้องเตรียมข้อมูล เพื่อใช้สำหรับงานการลงทะเบียนนักศึกษา โดยเลือกรายการที่ 1 จาก เมนูหลัก บนจอภาพจะปรากฏดังรูปที่ 4.2 เลือกรายการที่ 1 เพื่อเข้าสู่ระบบป้อนข้อมูลประวัติส่วนตัว ดังรูปที่ 4.6

ซึ่งมีการทำงาน 6 อย่างคือ

- " 1: ก่อน " เป็นการดูข้อมูลก่อนหน้าที่แสดงอยู่
- " 2: หลัง " เป็นการดูข้อมูลถัดไปจากข้อมูลที่แสดงอยู่
- " 3: ค้นหา " เป็นการค้นหาข้อมูลโดยใช้รหัสวิชาเป็นคีย์ในการค้นหา
- " 4: แก้ไข " เป็นการแก้ไขข้อมูลที่แสดงอยู่
- " 5: เพิ่มเติม " เป็นการเพิ่ม เติมเพิ่มข้อมูล เข้าสู่เพิ่มข้อมูล
- " 6: ลบ " เป็นการลบข้อมูลที่แสดงอยู่ออกจากเพิ่มข้อมูล

การเลือกการทำงานสามารถกระทำได้ 2 วิธี คือ เลือกโดยใช้ฟังก์ชันคีย์ ตามหมายเลขของหัวข้อการทำงาน เช่น กด F4 เมื่อต้องการแก้ไขข้อมูล หรือ ใช้ปุ่มลูกศร ซ้าย-ขวา เลื่อนแถบไปยังหัวข้อการทำงานที่ต้องการ และกด ENTER เพื่อทำงาน

รายการที่ 1 และ 2 เป็นการนำข้อมูลที่อยู่ก่อนหน้าและถัดไปขึ้นมาแสดงบนจอภาพ เมื่อถึงข้อมูลแรกและข้อมูลสุดท้าย จะมีข้อความบอกให้ทราบ

รายการที่ 3 ค้นหาข้อมูล จะมีกรอบให้ป้อนรหัสนักศึกษาที่ต้องการเรียกดูที่มุมซ้ายด้านล่างของจอภาพ จะมีข้อความ

F10 ตกลง

ESC ยกเลิก

เมื่อป้อนรหัสวิชาเรียนเรียบร้อยแล้ว กด F10 เพื่อทำการค้นหา หรือกด ESC เพื่อยกเลิกการค้นหา ถ้าทำการค้นหาสำเร็จ หรือพบรหัสนักศึกษาในเพิ่มข้อมูล ก็นำข้อมูลขึ้นแสดงบนจอภาพ แต่ถ้าไม่พบรหัสนักศึกษาในเพิ่มข้อมูลก็จะมีข้อความบอกว่าไม่พบรหัสนักศึกษาในเพิ่มข้อมูล

รายการที่ 4 แก้ไข มีรายละเอียดดังนี้

- ใช้ปุ่มลูกศรขึ้น-ลง เพื่อเลื่อนการการแก้ไขไปยังส่วนต่าง ๆ
- กดปุ่ม Insert เมื่อต้องการป้อนข้อมูลแบบแทรกค่าและกดอีกครั้งเมื่อต้องการป้อนข้อมูลแบบทับข้อมูลเดิม
- ในส่วนข้อมูลที่เป็นตัวเลขจะมีการป้องกัน โดยถ้าป้อนข้อมูลที่ไม่ใช่ตัวเลขจะมีการแสดงข้อความบอกข้อผิดพลาดและให้ป้อนข้อมูลนั้นใหม่
- กด F10 เมื่อต้องการเก็บข้อมูลที่แก้ไขเรียบร้อยแล้วหรือกด ESC เมื่อต้องการยกเลิกการแก้ไขทั้ง 2 กรณี การทำงานจะกลับไปยังการเลือกการทำงาน

รายการที่ 5 เพิ่มเติม จะมีรายละเอียดการทำงานคล้ายกับการแก้ไข ต่างกันที่ การแก้ไขข้อมูล ใน Field ต่างๆ ยังคงอยู่แต่การเพิ่มเติมจะเป็นการป้อนข้อมูลใหม่ทั้งหมด

รายการที่ 6 ลบ เมื่อต้องการลบข้อมูลที่แสดงอยู่ โดยมีข้อความถามยืนยัน การลบข้อมูล ถ้าต้องการลบ กด F10 ถ้าต้องการยกเลิกการลบข้อมูล กด ESC

เนื่องจากแฟ้มข้อมูลประวัติส่วนตัว มีการเรียกใช้ในระบบอื่น จึงควรมีการตรวจสอบความถูกต้องก่อน ที่จะทำการลบ เพราะมีผลกระทบกับระบบอื่น ๆ

เมื่อป้อนข้อมูล โดยเลือกการทำงานเพิ่มเติม แก้ไข จนข้อมูลถูกต้องเรียบร้อยแล้ว ออกจากการป้อนข้อมูลประวัติส่วนตัว โดยการเลื่อนแถบไปที่ "9 : EXIT" กด Enter หรือกด F9 จะปรากฏรายการทำงานของระบบประวัติ ตามรูป 4.2 กด F9 อีกครั้งเพื่อออกสู่รายการเมนูหลัก ดังรูป 4.1

เลือกรายการที่ 2 เพื่อทำการป้อนข้อมูลประวัติการศึกษา บนจอภาพจะปรากฏดังรูป 4.7 การทำงานเหมือนกับระบบป้อนข้อมูลประวัติส่วนตัว คือ ประกอบด้วย การทำงาน 6 อย่าง คือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูล แก้ไขข้อมูล เพิ่มเติมข้อมูล และลบข้อมูล โดยมีข้อแตกต่างดังนี้

- การป้อนข้อมูลเริ่มจากใส่รหัสนักศึกษาที่ต้องการ เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลอยู่หรือไม่ ถ้าพบจะปรากฏชื่อภาษาไทย ของนักศึกษาคนนั้น ใน Field ชื่อและชื่อภาษาไทยของภาควิชาใน Field ของภาควิชา ถ้าไม่พบจะแจ้งให้ทราบว่าไม่พบรหัสนักศึกษาในแฟ้มข้อมูล และให้ป้อนรหัสใหม่หรือยกเลิก

การออกจากรายการประวัติการศึกษา เหมือนกับรายการประวัติส่วนตัว คือ ให้กด F9 หรือ

โดยการเลื่อนแถบไปที่ "9 : EXIT" กด Enter จะปรากฏรายการการทำงานของประวัติ

นักศึกษา ดังรูป 4.2

การบันทึกรายวิชา

การเตรียมข้อมูล

เนื่องจากระบบการทำงานทะเบียนนักศึกษามีความเกี่ยวข้องกับรายวิชาจึงต้องมีการเตรียมข้อมูลรายวิชาเพื่อใช้สำหรับการลงทะเบียนโดยเลือกรายการการทำงานที่ 2 จากรายการหลักบนจอภาพจะปรากฏรายการการทำงาน การบันทึกรายละเอียดของวิชาดังรูป 4.3 ต่อจากนั้นก็เลือกรายการการทำงานที่ 4 เพื่อบันทึกรายละเอียดของวิชาทั้งหมด บนจอภาพจะปรากฏดังรูป 4.11 การทำงานเหมือนกับระบบบ้อนข้อมูลประวัติส่วนตัวคือ ประกอบด้วยการทำงาน 6 อย่างคือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูลแก้ไขข้อมูล เพิ่มเติมข้อมูล และลบข้อมูล

การออกจากรายการการบันทึกรายละเอียดของวิชาทั้งหมดให้กด F9 หรือเลื่อนแถบแสงไปที่ "9:EXIT" กด ENTER จะปรากฏรายการการทำงานของการบันทึกรายวิชา ดังรูป 4.3

เลือกรายการที่ 1 เพื่อทำการบ้อนข้อมูลรายวิชาบังคับเฉพาะภาคการศึกษาบนจอภาพจะปรากฏดังรูป 4.8 การทำงานเหมือนกับการบันทึกรายละเอียดของวิชาทั้งหมด คือ ประกอบด้วยการทำงาน 6 อย่าง คือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูล แก้ไขข้อมูล เพิ่มเติมข้อมูล และลบข้อมูล โดยมีข้อแตกต่างคือ

- การบ้อนข้อมูลเริ่มจากภาควิชา เลือกภาควิชาโดยใช้ปุ่มลูกศร ซ้าย-ขวา หรือแป้นคานเคาะวรรค (space bar) เป็นตัวเลือก
- ภาคเรียนจะเป็นการเลือกภาคเรียนโดยใช้ปุ่มลูกศรซ้าย-ขวา หรือแป้นคานเคาะวรรค (space bar) เป็นตัวเลือก
- การบ้อนรหัสวิชา เมื่อบ้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ถ้าพบจะปรากฏชื่อภาษาไทยใน field ของชื่อวิชา จำนวนหน่วยกิตใน field ของหน่วยกิต จำนวนทฤษฎี และปฏิบัติใน field ของทฤษฎี และปฏิบัติตามลำดับ

เมื่อบ้อนข้อมูลเสร็จแล้วกด F10 เพื่อจัดเก็บข้อมูลลงแฟ้มข้อมูล ถ้าไม่ต้องการจัดเก็บให้กด ESC การออกจากรายการหมวดวิชาเลือกเฉพาะภาคการศึกษา เหมือนกับหมวดวิชาบังคับเฉพาะภาคการศึกษา คือให้กด F9 หรือโดยการเลื่อนแถบไปที่ "9:EXIT" กด ENTER จะ

เลือกรายการที่ 2 เพื่อทำการป้อนข้อมูลรายวิชาบังคับเฉพาะภาคการศึกษา บนจอภาพจะปรากฏดังรูป การทำงานเหมือนกับการบันทึกหมวดวิชาบังคับเฉพาะภาคการศึกษา คือประกอบด้วยการทำงาน 6 อย่าง คือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูล แก้ไขข้อมูล เพิ่มเติมข้อมูลและ ลบข้อมูล

- การป้อนข้อมูลภาควิชา
- การป้อนข้อมูลภาคเรียน
- การป้อนข้อมูลวิชาเลือก เป็นการเลือกวิชาเลือกโดยใช้ปุ่มลูกศร ซ้าย-ขวาหรือแป้นคานเคาะวรรค (space bar) เป็นตัวเลือก

- การป้อนข้อมูลรหัสวิชา เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ถ้าพบ จะปรากฏชื่อวิชาภาษาไทยใน field ชื่อวิชาและจำนวนหน่วยกิตใน field หน่วยกิต

เมื่อป้อนข้อมูลเสร็จแล้วกด F10 เพื่อจัดเก็บข้อมูลลงแฟ้มข้อมูล ถ้าไม่ต้องการจัดเก็บให้กด ESC การออกจากรายการหมวดวิชาเลือกเฉพาะภาคการศึกษา เหมือนกับหมวดวิชาบังคับเฉพาะภาคการศึกษา คือให้กด F9 หรือโดยการเลื่อนแถบไปที่ "9:EXIT" กด ENTER จะปรากฏรายการการทำงาน การบันทึกรายวิชาดังรูป 4.3

เลือกรายการที่ 3 เพื่อทำการป้อนข้อมูล รายละเอียดวิชาที่เปิดสอนในปัจจุบัน บนจอภาพจะปรากฏดังรูป 4.10 การทำงานเหมือนกับการบันทึกหมวดวิชาเลือกเฉพาะภาคการศึกษา คือประกอบด้วยการทำงาน 6 อย่าง คือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูล แก้ไขข้อมูล เพิ่มเติมข้อมูล และลบข้อมูล

- การป้อนข้อมูลรหัสวิชา เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูล หรือไม่ถ้าพบจะปรากฏชื่อวิชาภาษาไทย ชื่อวิชาภาษาอังกฤษใน field ชื่อวิชา และชื่อวิชา (อังกฤษ) ตามลำดับ
- การป้อนข้อมูลวันสอน เวลาเรียน และช่วงการสอบ จะเป็นการเลือกโดยใช้ปุ่มลูกศร ซ้าย-ขวา หรือแป้นเคาะวรรค(space bar) เป็นตัวเลือก

เมื่อป้อนข้อมูลเสร็จแล้ว ถ้าต้องการจัดข้อมูลลงแฟ้มข้อมูลให้กด F10 ถ้าไม่ต้องการจัดเก็บให้กด ESC

การออกจากรายการรายละเอียดวิชาทั้งหมด ก็เหมือนกับรายการอื่น ๆ คือให้กด F9 หรือ

เลื่อนแถบไปที่ "9:EXIT" กด ENTER จะปรากฏรายการการทำงานของการบันทึกรายวิชา เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดั่งรูป 4.3

เลือกรายการที่ 5 โครงสร้างหลักสูตรบนจอภาพจะปรากฏดั่งรูป 4.12 การป้อนข้อมูลจะป้อนภาควิชา ภาคเรียน ชั้นปี และรหัสวิชาที่ต้องการ เมื่อใส่รหัสวิชาแล้วจะทำการค้นหาวิชานั้นจากแฟ้มข้อมูลการเปิดสอนรายวิชา ถ้าพบจะแสดงรายละเอียด ชื่อวิชา จำนวนหน่วยกิต ทฤษฎี-ปฏิบัติ ถ้าไม่พบจะแจ้งข้อความให้ทราบและทำการป้อนรหัสวิชาใหม่เมื่อป้อนข้อมูลเสร็จแล้วต้องการจัดเก็บข้อมูลลงแฟ้มข้อมูลให้กด F10 ถ้าไม่ต้องการจัดเก็บให้กด ESC จะออกจากระบบจะปรากฏการทำงานของการทำงานบันทึกกรายวิชา ดั่งรูป 4.3

การลงทะเบียนนักศึกษา

การเตรียมข้อมูล

ระบบการลงทะเบียนนักศึกษา มีความเกี่ยวข้องกับการกำหนดช่วงวันลงทะเบียน จะมีการบอกขอบเขตของการลงทะเบียน โดยการเลือกรายการที่ 3 จากเมนูหลักบนจอภาพ จะปรากฏดั่งรูป 4.4 เลือกรายการที่ 1 บนหน้าจอจะปรากฏดั่งรูป 4.13 เพื่อเข้าสู่ระบบป้อนข้อมูล กำหนดช่วงวันลงทะเบียน การทำการประกอบด้วย 3 รายการคือ ดูข้อมูลหลังสุด แก้ไขและการเพิ่มเติมข้อมูล

- การป้อนข้อมูล จะป้อนในลักษณะ วัน/เดือน/ปี ถ้าป้อนข้อมูลผิดไปจากความจริง โปรแกรมจะบอกความผิดพลาด เพื่อให้ป้อนข้อมูลใหม่

เลือกรายการที่ 2 สรุปผลการลงทะเบียน เป็นระบบป้อนข้อมูลการลงทะเบียน บนจอภาพจะปรากฏดั่งรูป 4.14 ดั่งมีรายการทำงานดังนี้

"1:เพิ่มเติม" เพิ่มเติมรายวิชาที่ต้องการลงทะเบียนเพิ่ม

"2:ลบ" ก่อนการลงทะเบียนที่ลงทะเบียนไปแล้ว

"3:นักศึกษาอื่น" ทำการลงทะเบียนของนักศึกษาแต่ละคน

การทำงานเริ่มจากรายการที่ 3:นักศึกษาอื่น ทำการป้อนรหัสนักศึกษา ภาควิชา ภาคเรียนที่ ชั้นปีที่ ที่ต้องการ เมื่อป้อนเสร็จกด ENTER โปรแกรมจะทำการตรวจสอบ ถ้าไม่มีข้อผิดพลาด โปรแกรมจะนำโครงสร้างหลักสูตรที่ตรงกับข้อมูลของนักศึกษาคนนั้นมาแสดง โดยไม่ต้องป้อนข้อมูลอีก แล้วทำการเพิ่ม-ถอน รายวิชาตามที่นักศึกษาได้ลงทะเบียน ซึ่งโปรแกรมจะทำการเก็บข้อมูลทุกครั้งทั้งลงทะเบียน และทำการเรียงลำดับรายวิชาเพื่อให้ง่ายต่อการตรวจสอบ

ผลการเรียนของนักศึกษา

การเตรียมข้อมูล

ผลการเรียนเป็นการทำงานขั้นตอนสุดท้ายของระบบงานทะเบียนนักศึกษา โดยการเลือกรายการที่ 4 จากเมนูหลักบนจอภาพจะปรากฏดังรูป 4.5 เลือกรายการที่ 1 เพื่อเข้าสู่ระบบการบันทึกเกรดบนหน้าจอ จะปรากฏดังรูป 4.15 เมื่อการสอบประจำภาคเรียนเสร็จสิ้นลง อาจารย์ประจำวิชาจะทำการแจ้งผลการสอบมายังฝ่ายประมวลผล การป้อนข้อมูลเริ่มจากการป้อนรหัสนักศึกษา ภาควิชา ภาคเรียนที่ ชั้นปีที่ โปรแกรมจะนำข้อมูลลงทะเบียนขึ้นมาแสดงบนจอภาพ จากนั้นป้อนข้อมูลผลการเรียนเสร็จเรียบร้อยกด F10 เพื่อเก็บข้อมูลผลการเรียนนั้น หรือกด ESC เพื่อยกเลิกการป้อนข้อมูลทั้งหมด โปรแกรมจะทำการคำนวณ ผลการศึกษาเฉลี่ยให้ทราบด้วย

เลือกรายการที่ 2 การแก้ไขเกรด เป็นการแก้ไขเกรดเฉพาะรายวิชาที่มีปัญหาบนจอภาพจะปรากฏดังรูป 4.16

- การป้อนข้อมูลรหัสวิชา เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ ถ้าพบจะปรากฏชื่อวิชาภาษาไทย ชื่ออาจารย์ผู้สอนใน field ของชื่อวิชา และอาจารย์ผู้สอนตามลำดับ
- การป้อนรหัสนักศึกษา เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ถ้าพบจะปรากฏชื่อวิชาภาษาไทย ภาควิชาภาษาไทย เกรด เกรดที่แก้ไขใน field ของ ชื่อภาควิชา เกรด และ เกรดที่แก้ไข ตามลำดับ

เลือกรายการที่ 3 สรุปผลเฉพาะภาคการศึกษา บนจอภาพจะปรากฏดังรูป 4.17

- การป้อนข้อมูลรหัสนักศึกษา เมื่อป้อนเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ถ้าพบจะปรากฏ ชื่อภาษาไทยใน field ของ ชื่อ-สกุล ทำการป้อนข้อมูลภาควิชา ภาคเรียนที่ ชั้นปีที่ เมื่อป้อนเสร็จจะปรากฏจำนวนหน่วยกิต เกรดที่ได้เฉพาะเทอมนี้ ของเดิมของใหม่ ตามลำดับ

คณะวิทยาศาสตร์
ระบบทะเบียนนักศึกษา (ฝ่ายทะเบียน)
1. ประวัตินักศึกษา 2. การบันทึกรายวิชา 3. การลงทะเบียนนักศึกษา 4. ผลการศึกษาของนักศึกษา 9. ออกจากระบบ

กด F1- F4 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 4.1- แสดงหน้าจอเมนูหลักของฝ่ายทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
ประวัตินักศึกษา
1. การบันทึกประวัติส่วนตัว 2. การบันทึกประวัติการศึกษา 9. ออกสู่มายการหลัก

กต F1- F2 หรือ ลูกศรขึ้น-ลง และกต RETURN เพื่อเลือกการทำงาน

รูปที่ 4.2 แสดงหน้าจอเมนูของประวัตินักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
การบันทึกรายวิชา 1. หมวดวิชาบังคับเฉพาะภาคการศึกษา 2. หมวดวิชาเลือกเฉพาะภาคการศึกษา 3. รายละเอียดวิชาที่เปิดสอนในปัจจุบัน 4. รายละเอียดวิชาทั้งหมด 5. โครงสร้างหลักสูตร 9. ออกสู่รายการหลัก

กด F1- F5 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 4.3 แสดงหน้าจอเมนูของการบันทึกรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
การลงทะเบียนนักศึกษา 1. กำหนดช่วงวันทะเบียน 2. สรุปผลการลงทะเบียน 9. ออกสูัรรายการหลัก

กด F1- F2 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 4.4 แสดงหน้าจอเมนูของการลงทะเบียนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
ผลการเรียนของนักศึกษา 1. การบันทึกเกรด 2. การแก้ไขเกรด 3. สรุปผลเฉพาะภาคการศึกษา 4. สรุปผลทั้งหมดที่ผ่านมา 9. ออกสัรยการหลัก

กค F1- F4 หรือ ลูกศรขึ้น-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 4.5 แสดงหน้าจอเมนูผลการเรียนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา: 99-999999

ภาควิชา: XXXXXXXXXXXXXXXXXX

ชื่อ(ไทย): XXXXXXXXXXXXXXXXXX

เพศ: XXX

นามสกุล(ไทย): XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME: XXXXXXXXXXXXXXXXXXXX

SURNAME: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด: 99/99/99

[dd/mm/yy]

วันเข้ารับการศึกษ: 99/99/99

[dd/mm/yy]

อ.ที่ปรึกษา: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

ชื่อผู้ปกครอง: XXXXXXXXXXXXXXXXXXXX

นามสกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

หน่วยกิตสะสม: 999

1: ก่อน 2: หลัง 3: ค้นหา 4: แก้ไข 5: เพิ่มเติม 6: ลบ

9: Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.6 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลประวัตินักศึกษา		
รหัสนักศึกษา: 99-999999		
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
ภาควิชา: XXXXXXXXXXXXXXXXXXXX		
ระดับการศึกษา	สถานศึกษา	ช่วงปีการศึกษา
ประถมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
ประถมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999

1: ก่อน 2: หลัง 3: ค้นหา 4: แก้ไข 5: เพิ่มเติม 6: ลบ 9: Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.7 แสดงหน้าจอของประวัติการศึกษาของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวดวิชาบังคับ	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	
ชั้นปีที่: 9	ภาคเรียนที่: 9
รหัสวิชา: 99-999	
ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX	
หน่วยกิต: 9	
ทฤษฎี: 9	ปฏิบัติ: 9

1: ก่อน 2: หลัง 3: แก้วไข 4: เติมเต็ม 5: ลบ 9: Exit

กด F1 - F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.8 แสดงหน้าจอของรายละเอียดวิชาบังคับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาเลือกเฉพาะภาคการศึกษา	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	ภาคเรียนที่: 9
วิชาเลือก: XXXXXXXXXXXXXXXXXXXX รหัสวิชา: 99-999 ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX หน่วยกิต: 9	

1:ก่อน 2:หลัง 3:แก้ไข 4:เพิ่มเติม 5:ลบ 9:Exit

กด F1 - F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.9 แสดงหน้าจอของรายละเอียดวิชาเลือกเฉพาะภาคการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลกระบวนวิชาที่เปิดสอนในเทอมปัจจุบัน

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ): XXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันสอน: 9

เวลาเรียน: 9

วันสอบ: 99/99 [dd/mm]

ช่วง: 9

1:ก่อน

2:หลัง

3:ค้นหา

4:แก้ไข

5:เพิ่มเติม

6:ลบ

9:Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.10 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนในเทอมนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายชื่อวิชาทั้งหมดในหลักสูตร

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ): XXXXXXXXXXXXXXXXXXXX

หน่วยกิต: 9

ทฤษฎี: 9

ปฏิบัติ: 9

วิชาก่อนหน้า: 99-999 99-999 99-999

1: ก่อน

2: หลัง

3: ค้นหา

4: แก้ไข

5: เพิ่มเติม

6: ลบ

9: Exit

กด F1 - F6 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.11 แสดงหน้าจอของรายละเอียดวิชาที่เปิดสอนทั้งหมดในหลักสูตร

โครงสร้างการศึกษา			
ภาควิชา: XXXXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
รหัสวิชา	ชื่อวิชา	หน่วยกิต	ทฤษฎี-ปฏิบัติ
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9
99-999	XXXXXXXXXXXXXXXXXXXXX	9	9 - 9

1:ก่อน 2:หลัง 3:แก้ไข 4:เพิ่มเติม 5:ลบ 9:Exit
 กด F1 - F5 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.12 แสดงหน้าจอของ โครงสร้างหลักสูตรทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดช่วงวันลงทะเบียน	
การลงทะเบียนปกติ	
เริ่มต้น: 99/99/99	[dd/mm/yy]
สิ้นสุด: 99/99/99	[dd/mm/yy]
กรณีเพิ่มหรือถอน	
เริ่มต้น: 99/99/99	[dd/mm/yy]
สิ้นสุด: 99/99/99	[dd/mm/yy]

1:หลัง 2:แก้ไข 3:เพิ่มเติม 9:Exit
 กด F1 – F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.13 แสดงหน้าจอของการกำหนดช่วงของการลงทะเบียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลการลงทะเบียน รหัสนักศึกษา: 99-999999 ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX ภาควิชา: XXXXXXXXXXXXXXXX ภาคเรียนที่: 9 ชั้นปีที่: 9		
รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXXX	9

รวม : 999

1:เพิ่มเติม 2:ลบ 3:นักศึกษาอื่น 9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.14 แสดงหน้าจอของการลงทะเบียนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลผลการศึกษาประจำภาคการศึกษา			
รหัสนักศึกษา: 99-999999			
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
ภาควิชา: XXXXXXXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
รหัสวิชา	ชื่อวิชา	หน่วยกิต	ผลการศึกษา
99-999	XXXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXXX	9	XX

รวม : 999 เฉลี่ย : 9.999

1: ผลการศึกษาของนักศึกษาอื่น

9:Exit

Esc-ยกเลิก

F10-ตกลง

รูปที่ 4.15 แสดงหน้าจอเมื่อเราต้องการป้อนผลการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแก้ไขเฉพาะรายวิชา	
รหัสวิชา: 99-999	
ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX	
อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
รหัสนักศึกษา: 99-999999	
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	
เกรด: XX	
เกรดที่แก้ไข: XX	
เกรดปัจจุบัน: XX	

1: ก่อน 2: หลัง 3: ค้นหา 4: แก้ไข 5: เพิ่มเติม 6: ลบ 9: Exit

กด F1 - F6 หรือ ลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.16 แสดงหน้าจอของการแก้ไขเฉพาะรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการศึกษาเฉพาะภาคเรียน			
รหัสนักศึกษา: 99-999999			
ชื่อ-สกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
ภาควิชา: XXXXXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
	เฉพาะเทอมนี้	ของเดิม	ของใหม่
จำนวนหน่วยกิต	999	999	999
เกรดที่ได้ (GPS/GPA)	9.999	9.999	9.999

1: ก่อน 2: หลัง 3: นักศึกษาอื่น

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.17 แสดงหน้าจอของการสรุปผลการศึกษาเฉพาะภาคเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การใช้งานโปรแกรมระบบฝ่ายอาจารย์

สำหรับอาจารย์ เมื่อเลือกใช้งาน TE.EXE และข้อรหัสผ่านเข้าสู่งานเรียบร้อยแล้ว จะเข้าสู่งานหลักตามรูปที่ 4.18 ซึ่งประกอบด้วย

1. เกรดของนักศึกษา
2. การตรวจสอบการลงทะเบียน
3. ข้อมูลนักศึกษา

เกรดของนักศึกษาประกอบด้วย

- การบันทึกเกรดเฉพาะรายวิชา
- ผลการศึกษาของนักศึกษาเฉพาะราย

การตรวจสอบผลการลงทะเบียน

- ข้อมูลการลงทะเบียน

ข้อมูลนักศึกษา

- ประวัติส่วนตัว
- ประวัติการศึกษา

ขั้นตอนการทำงาน

เกรดของนักศึกษา

การเตรียมข้อมูล

ระบบการทำงานทะเบียนนักศึกษา ส่วนที่เกี่ยวข้องกับอาจารย์คือ เกรดของนักศึกษา โดยเลือกจากรายการที่ 1 จากรายการหลักบนจอภาพ จะปรากฏดังรูป 4.18 เลือกรายการที่ 1 จากรูป 4.19 เพื่อเข้าสู่ระบบการบันทึเกรดเฉพาะรายวิชา บนหน้าจอภาพจะปรากฏดังรูป 4.20

การทำงานจะเหมือนกับประวัติส่วนตัวของนักศึกษา ในส่วนของทะเบียน คือดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป ค้นหาข้อมูล แก้ไขข้อมูล เพิ่มเติมข้อมูล และลบข้อมูล

- การบันทึข้อมูลรหัสวิชา เมื่อบันทึเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ ถ้ามีจะปรากฏ ชื่อวิชาภาษาไทย ชื่ออาจารย์ผู้สอน ใน field ชื่อวิชา และอาจารย์ผู้สอนตามลำดับ

- การบันทึข้อมูลรหัสวิชา เมื่อบันทึเสร็จโปรแกรมจะทำการตรวจสอบว่ามีข้อมูลหรือไม่ ถ้าเอกสารนี้เป็นเอกสารลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ใช้ได้เห็นใบโฆษณาเกี่ยวกับการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

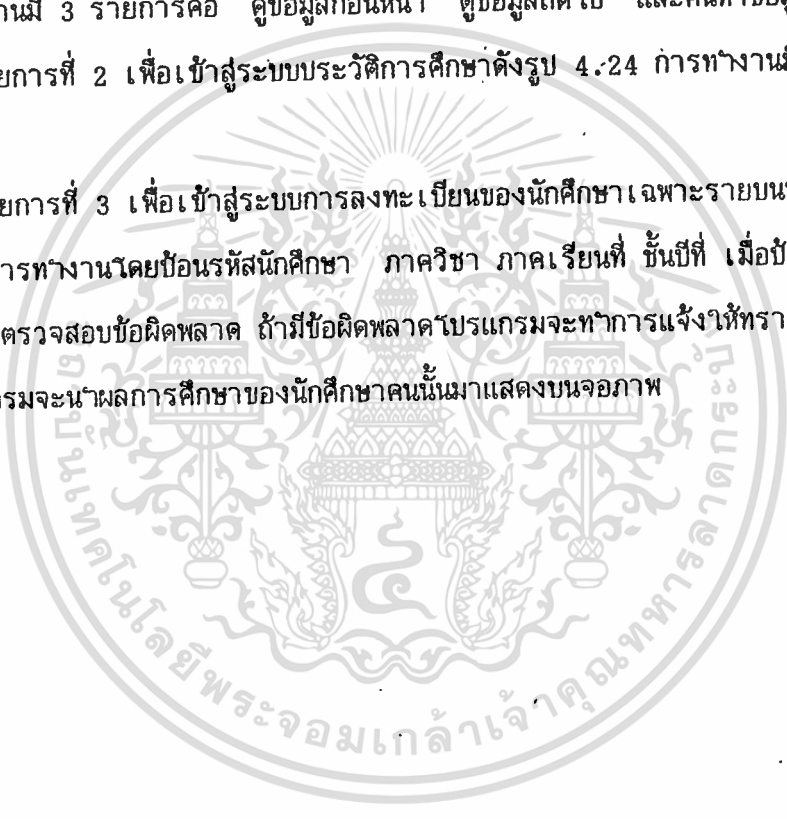
มีจะปรากฏ ชื่อภาษาไทย ภาควิชาภาษาไทยใน field ของชื่อ และภาควิชาตามลำดับ
เมื่อเลือกรายการที่ 2 จะปรากฏผลการศึกษาเฉพาะรายของนักศึกษา เป็นการตรวจสอบ
ข้อมูลเพียงอย่างเดียว หน้าจอจะปรากฏดังรูป 4.21

ข้อมูลนักศึกษา

เป็นการเรียกข้อมูลเพื่อดูเพียงอย่างเดียว ไม่มีการเพิ่มเติมหรือแก้ไข เมื่อเลือกรายการ
ที่ 3 จากเมนูหลักบนจอภาพจะปรากฏดังรูป 4.22 เลือกรายการที่ 1 เพื่อเข้าสู่ระบบข้อมูลดังรูป
4.23 การทำงานมี 3 รายการคือ ดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป และค้นหาข้อมูล

เลือกรายการที่ 2 เพื่อเข้าสู่ระบบประวัติการศึกษาดังรูป 4.24 การทำงานมี 3 รายการ
เหมือนกัน

เลือกรายการที่ 3 เพื่อเข้าสู่ระบบการลงทะเบียนของนักศึกษาเฉพาะรายบนหน้าจะปรากฏ
ดังรูป 4.25 การทำงานโดยป้อนรหัสนักศึกษา ภาควิชา ภาคเรียนที่ ชั้นปีที่ เมื่อป้อนเสร็จโปรแกรม
จะทำการตรวจสอบข้อผิดพลาด ถ้ามีข้อผิดพลาดโปรแกรมจะทำการแจ้งให้ทราบ ถ้าไม่มีข้อ
ผิดพลาดโปรแกรมจะนำผลการเรียนของนักศึกษาคนนั้นมาแสดงบนจอภาพ



คณะวิทยาศาสตร์
ระบบทะเบียนนักศึกษา (อาจารย์)
1. เกรดของนักศึกษา
2. ผลการลงทะเบียนเฉพาะราย
3. ข้อมูลของนักศึกษา
9. ออกจากระบบ

กด F1 - F3 หรือ ลูกศรบน-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 4.18 แสดงรายการงานของส่วนของอาจารย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
เกรดของนักศึกษา 1. การบันทึกเกรดเฉพาะรายวิชา 2. ผลการศึกษาของนักศึกษาเฉพาะราย 9. ออกสู่อายการหลัก

กค F1 - F2 หรือ ลูกศรบน-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 4.19 แสดงงานในส่วนของเกรดนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกเกรดเฉพาะรายวิชา

รหัสวิชา : XX-XXXX

ชื่อวิชา : XXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน : XXXXXXXXXXXXXXXXXXXX

รหัสนักศึกษา : 99-999999

ชื่อ : XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา : XXXXXXXXXXXXXXXXXXXX

เกรด : XXX

เกรดแก้ไข : XXX

1:ก่อน 2:หลัง 3:ค้นหา 4:แก้ไข 5:เพิ่มเติม 6:ลบ 9:Exit

กด F1 - F6 หรือ ลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.20 แสดงการทำงานในส่วนของการเปลี่ยนแปลงแก้ไขผลการศึกษาเป็นรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการเรียนของนักศึกษาเฉพาะราย			
รหัสนักศึกษา: 99-999999			
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
ภาควิชา: XXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
รหัสวิชา	ชื่อวิชา	หน่วยกิต	ผลการเรียน
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX
99-999	XXXXXXXXXXXXXXXXXXXX	9	XX

รวม : 999

เฉลี่ย : 9.999

1:ผลการเรียนของนักศึกษาอื่น

9:Exit

Esc-ยกเลิก

F10-ตกลง

รูปที่ 4.21 แสดงหน้าของการเปลี่ยนแปลงแก้ไขผลการเรียนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
ข้อมูลนักศึกษา 1. ประวัติส่วนตัว 2. ประวัติการศึกษา 9. ออกสูรายการหลัก

กค F1 - F2 หรือ ลูกศรบน-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 4.22 แสดงหน้าของการตรวจสอบข้อมูลของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา: 99-999999

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อ (ไทย): XXXXXXXXXXXXXXXXXXXX เพศ: XXX

นามสกุล (ไทย): XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME: XXXXXXXXXXXXXXXXXXXX

SURNAME: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด: 99/99/99 [dd/mm/yy]

วันเข้ารับการศึกษ: 99/99/99 [dd/mm/yy]

อ.ที่ปรึกษา: XXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

ชื่อผู้ปกครอง: XXXXXXXXXXXXXXXXXXXX

นามสกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

หน่วยกิตสะสม: 999

1: ก่อน 2: หลัง 3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.23 แสดงหน้าจอของประวัติส่วนตัวนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลประวัตินักศึกษา		
รหัสนักศึกษา: 99-999999		
ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX		
ภาควิชา: XXXXXXXXXXXXXXXXXXXX		
ระดับการศึกษา	สถานศึกษา	ช่วงปีการศึกษา
ประถมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
ประถมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนต้น	XXXXXXXXXXXXXXXXXXXX	9999-9999
มัธยมศึกษาตอนปลาย	XXXXXXXXXXXXXXXXXXXX	9999-9999

1: ก่อน 2: หลัง 3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.24 แสดงหน้าจอของประวัติการศึกษาของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลการลงทะเบียนของนักศึกษาเฉพาะราย รหัสนักศึกษา: 99-999999 ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX ภาควิชา: XXXXXXXXXXXXXXXX ภาคเรียนที่: 9 ชั้นปีที่: 9		
รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9

รวม : 999

1:เพิ่มเติม 2:ลบ 3:นักศึกษาอื่น 9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.25 แสดงหน้าจอตรวจสอบการลงทะเบียนของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การใช้งานโปรแกรมระบบฝ่ายนักศึกษา

สำหรับนักศึกษา เมื่อเลือกใช้งาน PRO.EXE และบอกรหัสผ่านเข้าสู่ระบบการทำงานเรียบร้อยแล้ว จะเข้าสู่การเลือกรายการการทำงานหลักตามรูป 4.26 ซึ่งประกอบด้วย

1. ประวัติส่วนตัว
2. ตรวจสอบผลการศึกษา
3. ลงทะเบียนและเพิ่มถอนวิชาเรียน
4. ตรวจสอบรายละเอียดรายวิชาที่เปิดสอน

ตรวจสอบผลการศึกษาประกอบด้วย

- ตรวจสอบเป็นรายวิชา
- ตรวจสอบผลการศึกษาเฉพาะภาคเรียน

ตรวจสอบรายวิชาที่เปิดสอน

- วิชาทั้งหมดในหลักสูตร
- วิชาบังคับเฉพาะภาคการศึกษา
- วิชาเลือกเฉพาะภาคการศึกษา
- วิชาที่เปิดสอนในปัจจุบัน

ขั้นตอนการทำงาน

ประวัติส่วนตัว

ระบบงานทะเบียนนักศึกษา ส่วนที่เกี่ยวข้องการศึกษามีทั้งการขอข้อมูลเพียงอย่างเดียว และบอกรหัสผ่านเพื่อเพิ่มข้อมูล เมื่อเลือกรายการ 1 จากรายการเมนูหลักบนจอภาพจะปรากฏดังรูป 4.27 เพื่อให้บอกรหัสนักศึกษา เมื่อบอกรหัสนักศึกษาเสร็จโปรแกรมจะทำการตรวจสอบ ถ้ามีข้อมูล จะแสดงข้อมูลของนักศึกษาคนนั้น บนหน้าจอจะปรากฏดังรูป 4.28 ถ้าไม่มีข้อมูลโปรแกรมจะแจ้งให้ทราบ

ตรวจสอบผลการศึกษา

เป็นการดูข้อมูลผลการศึกษา เมื่อเลือกรายการ 2 จากรายการเมนูหลักบนจอภาพจะปรากฏดังรูป 4.29 เลือก รายการที่ 1 เพื่อเข้าสู่การตรวจสอบผลการศึกษาเฉพาะรายวิชาบนจอภาพจะปรากฏดังรูป 4.30

เลือกรายการที่ 2 เพื่อเข้าสู่การสรุปผลการศึกษาเฉพาะภาคเรียน บนหน้าจอจะปรากฏดังรูป 4.31

การลงทะเบียน

เป็นระบบป้อนข้อมูลการลงทะเบียน โดยเลือกรายการที่ 3 จากรายการหลักบนหน้าจอจะปรากฏดังรูป 4.32 ซึ่งมีรายการทำงานดังนี้

"1:เพิ่มเติม" เพิ่มเติมรายวิชาที่ต้องการลงทะเบียนเพิ่ม

"2:ลบ" ถอนการลงทะเบียนที่ลงทะเบียนไปแล้ว

"3:นักศึกษาอื่น" ทำการลงทะเบียนของนักศึกษาแต่ละคน

การทำงานเริ่มจากรายการ 3:นักศึกษาอื่น ทำการป้อนรหัสนักศึกษา ภาควิชา ภาคเรียนที่ขึ้นปีที่ ที่ต้องการ เมื่อป้อนเสร็จกด ENTER โปรแกรมจะทำการตรวจสอบ ถ้าไม่มีข้อผิดพลาด โปรแกรมจะนำโครงสร้างหลักสูตร ที่ตรงกับข้อมูลของนักศึกษาคนนั้นมาแสดง โดยไม่ต้องป้อนข้อมูลอีก แล้วทำการเพิ่ม-ถอน รายวิชาตามที่นักศึกษาได้ลงทะเบียนซึ่งโปรแกรมจะทำการเก็บข้อมูลทุกครั้งที่ลงทะเบียน และทำการเรียงลำดับรายวิชาเพื่อให้ง่ายต่อการตรวจสอบ

รายละเอียดของวิชา

เป็นการเรียกข้อมูลเดียวกับรายวิชา เพื่อศึกษาถึงรายละเอียดต่าง ๆ ของกระบวนวิชาในหลักสูตร โดยการเลือกรายการที่ 4 ของรายการเมนูหลัก จอภาพจะปรากฏดังรูป 4-33

เลือกรายการ 1 จะแสดงรายวิชาทั้งหมดในหลักสูตรบนหน้าจอจะปรากฏดังรูป 4-37 การทำงานมีรายการคือดูข้อมูลก่อนหน้า ดูข้อมูลถัดไป และค้นหาข้อมูล

เลือกรายการ 2 จะแสดงรายวิชาบังคับเฉพาะภาคการศึกษา บนหน้าจอจะปรากฏ ดังรูป 4-34 การทำงานมี 2 รายการคือ ดูข้อมูลก่อนหน้า และดูข้อมูลถัดไป

เลือกรายการ 3 จะแสดงรายวิชาเลือกเฉพาะภาคการศึกษา บนหน้าจอจะปรากฏดังรูป

4-35 การทำงานมี 2 รายการคือ ดูข้อมูลก่อนหน้า และดูข้อมูลถัดไป

เลือกรายการ 4 จะแสดงรายวิชาที่เปิดสอนในปัจจุบัน บนหน้าจอจะปรากฏดังรูป 4-36

การทำงานมี 3 รายการคือ ดูข้อมูลก่อนหน้า และดูข้อมูลถัดไป และค้นหาข้อมูล



คณะวิทยาศาสตร์

ระบบทะเบียนนักศึกษา (นักศึกษา)

1. ประวัติส่วนตัว
2. ตรวจสอบผลการศึกษา
3. ลงทะเบียนและเพิ่ม, ถอนวิชาเรียน
4. ตรวจสอบรายละเอียดวิชาที่เปิดสอน
9. ออกจากระบบ

กด F1 - F4 หรือ ลูกศรขึ้น-ลง และกด RETURN เพื่อเลือกการทำงาน

รูปที่ 4.26 แสดงรายการงานของส่วนนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น, ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บ่อนรหัสนักศึกษา

รหัสนักศึกษา : XX-XXXXXX

ESC-ยกเลิก

F10-ตกลง



รูปที่ 4.27 แสดงหน้าจอให้บ่อนรหัสนักศึกษาเพื่อให้เห็นประวัติของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติส่วนตัวนักศึกษา

รหัสนักศึกษา: 99-999999

ภาควิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อ(ไทย): XXXXXXXXXXXXXXXXXXXX

เพศ: XXX

นามสกุล(ไทย): XXXXXXXXXXXXXXXXXXXXXXXXXXXX

NAME: XXXXXXXXXXXXXXXXXXXX

SURNAME: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

วันเกิด: 99/99/99 [dd/mm/yy]

วันเข้ารับการศึกษ: 99/99/99 [dd/mm/yy]

อ.ที่ปรึกษา: XXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

ชื่อผู้ปกครอง: XXXXXXXXXXXXXXXXXXXX

นามสกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

ที่อยู่ปัจจุบัน: XXXXXXXXXXXXXXXXXXXXXXXXXXXX

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

โทรศัพท์: 999-9999

หน่วยกิตสะสม: 999

9:Exit

กด F1 - F1 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.28 แสดงประวัติของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
ตรวจสอบผลการศึกษา 1. ตรวจสอบเป็นรายวิชา 2. ตรวจสอบผลเป็นปีการศึกษา 9. ออกสู่มายการหลัก

กค F1 – F2 หรือ ลูกศรขึ้น-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 4.29 แสดงรายการย่อยของการตรวจสอบผลการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบผลการเรียนเฉพาะรายวิชา	
รหัสวิชา :	XX-XXX
ชื่อวิชา :	XXXXXXXXXXXXXXXXXXXX
ชื่อวิชา(อังกฤษ) :	XXXXXXXXXXXXXXXXXXXX
หน่วยกิต :	9
วันที่ลงทะเบียน :	99/99/99
เกรด :	XX

Search Window
รหัสวิชา: XX-XXX

1: ต้องการตรวจสอบวิชาอื่น

9: ไม่ต้องการ หรือ Exit

ESC-ยกเลิก

F10-ตกลง

กรุณาใส่รหัสวิชา

รูปที่ 4.30 แสดงการตรวจสอบผลการเรียนเป็นรายวิชา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการศึกษาเฉพาะภาคเรียน			
รหัสนักศึกษา: 99-999999			
ชื่อ-สกุล: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX			
ภาควิชา: XXXXXXXXXXXXXXXXXXXXX			
ภาคเรียนที่: 9		ชั้นปีที่: 9	
	เฉพาะเทอมนี้	ของเดิม	ของใหม่
จำนวนหน่วยกิต	999	999	999
เกรดที่ได้ (GPS/GPA)	9.999	9.999	9.999

1:นักศึกษาอื่น

9:Exit

กด F1 - F1 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.31 แสดงหน้าจอการตรวจสอบผลการศึกษาเป็นภาคการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบป้อนข้อมูลการลงทะเบียน

รหัสนักศึกษา: 99-999999

ชื่อ: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

ภาควิชา: XXXXXXXXXXXXXXXX

ภาคเรียนที่: 9 ชั้นปีที่: 9

รหัสวิชา	ชื่อวิชา	หน่วยกิต
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
99-999	XXXXXXXXXXXXXXXXXXXX	9
.....		
รวม:		999

1:เพิ่มเติมวิชา 2:ถอนวิชา 3:นักศึกษาอื่น 9:Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

รูปที่ 4.32 แสดงการลงทะเบียนเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิทยาศาสตร์
รายละเอียดของรายวิชา 1. วิชาทั้งหมดในหลักสูตร 2. วิชาบังคับเฉพาะภาคการศึกษา 3. วิชาเลือกเฉพาะภาคการศึกษา 4. วิชาที่เปิดสอนในปัจจุบัน 9. ออกสู่อายการหลัก

กค F1 – F4 หรือ ลูกศรขึ้น-ลง และกค RETURN เพื่อเลือกการทำงาน

รูปที่ 4.33 แสดงรายการย่อยของการตรวจสอบรายละเอียดวิชาเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาบังคับเฉพาะภาคการศึกษา	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	
ชั้นปีที่: 9	ภาคเรียนที่: 9
รหัสวิชา: 99-999	
ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX	
หน่วยกิต: 9	
ทฤษฎี: 9	ปฏิบัติ: 9



1: ก่อน
2: หลัง
9: Exit

กศ F1 – F2 หรือลูกศรขึ้น-ลง และกศ RETURN

รูปที่ 4.34 แสดงรายละเอียดวิชาเรียนที่เป็นวิชาบังคับของภาคการเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิชาเลือกเฉพาะภาคการศึกษา	
ภาควิชา: XXXXXXXXXXXXXXXXXXXX	ภาคเรียนที่: 9
วิชาเลือก: XXXXXXXXXXXXXXXXXXXX รหัสวิชา: 99-999 ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX หน่วยกิต: 9	

1: ก่อน

2: หลัง

9: Exit

กค F1 - F2 หรือลูกศรขึ้น-ลง และกค RETURN

รายวิชาที่เปิดสอนในเทอมปัจจุบัน

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ): XXXXXXXXXXXXXXXXXXXX

อาจารย์ผู้สอน: XXXXXXXXXXXXXXXXXXXX

วันสอน: 9

เวลาเรียน: 9

วันสอบ: 99/99 [dd/mm]

ช่วง: 9

Search Window

รหัสวิชา : 99-999

1: ก่อน

2: หลัง

3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

ESC-ยกเลิก

F10-ตกลง

รายละเอียดวิชาเปิดสอนในหลักสูตร

รหัสวิชา: 99-999

ชื่อวิชา: XXXXXXXXXXXXXXXXXXXX

ชื่อวิชา(อังกฤษ): XXXXXXXXXXXXXXXXXXXX

หน่วยกิต: 9

ทฤษฎี: 9

ปฏิบัติ: 9

วิชาก่อนหน้า: 99-999 99-999 99-999

Search Window

รหัสวิชา : 99-999

1: ก่อน 2: หลัง 3: ค้นหา

9: Exit

กด F1 - F3 หรือลูกศรขึ้น-ลง และกด RETURN

ESC-ยกเลิก

F10-ตกลง

บทที่ 5 สรุปผลและข้อเสนอแนะ

สรุปผล

งานระบบทะเบียนเดิมเป็นการกระทำด้วยมือ (ไม่ใช้คอมพิวเตอร์) เป็นส่วนใหญ่ ระบบงานจึงมีความซับซ้อนยุ่งยากมาก ทำให้งานทะเบียนในปัจจุบันมีความล่าช้ามาก การนำระบบคอมพิวเตอร์เข้ามาช่วยนั้น จะทำให้ระบบงานทะเบียนมีประสิทธิภาพมากขึ้น โดยจะทำให้ระบบงานมีความคล่องตัวรวดเร็วมากขึ้น และข้อมูลถูกเก็บไว้ในฐานข้อมูลเดียวกัน ทำให้ง่ายต่อการปรับปรุงและเปลี่ยนแปลง และจากการที่เรานำระบบ LAN เข้ามาประยุกต์ใช้ในระบบงานทะเบียนนั้นทำให้สามารถเรียกใช้โปรแกรมได้ในเวลาพร้อมๆ กัน จึงทำให้เกิดความรวดเร็วในการลงทะเบียนมากขึ้น

งานระบบทะเบียนที่ใช้ คอมพิวเตอร์กับ LAN เข้ามาช่วยนั้น เราได้แบ่งการทำงานออกเป็น 3 ส่วนใหญ่ ๆ คือ

1. ส่วนของฝ่ายทะเบียน
2. ส่วนของฝ่ายอาจารย์
3. ส่วนของฝ่ายนักศึกษา

เนื่องจากในระบบงานมีการแบ่งงานออกเป็นแผนก และระบบที่พัฒนาขึ้นมีการออกแบบระบบฐานข้อมูลให้เป็นฐานข้อมูลรวมศูนย์กลาง ดังนั้นจึงสามารถที่จะนำเอาระบบโครงข่ายเข้ามาใช้งานได้ โดยมีส่วประมวลผลกลาง และฐานข้อมูล เป็นส่วนบริการไปยังหน่วยงานต่าง ๆ ที่ต้องการใช้ข้อมูลร่วมกัน เป็นการประหยัดในการเก็บข้อมูลและข้อมูลมีความเชื่อถือได้ เพราะมีการจัดเก็บในที่เดียว การแก้ไขต่าง ๆ จะกระทำในฐานข้อมูลกลาง และสะดวกในการขยายระบบการทำงานไปยังแผนกอื่น ๆ ในอนาคต

ส่วนการนำเข้า

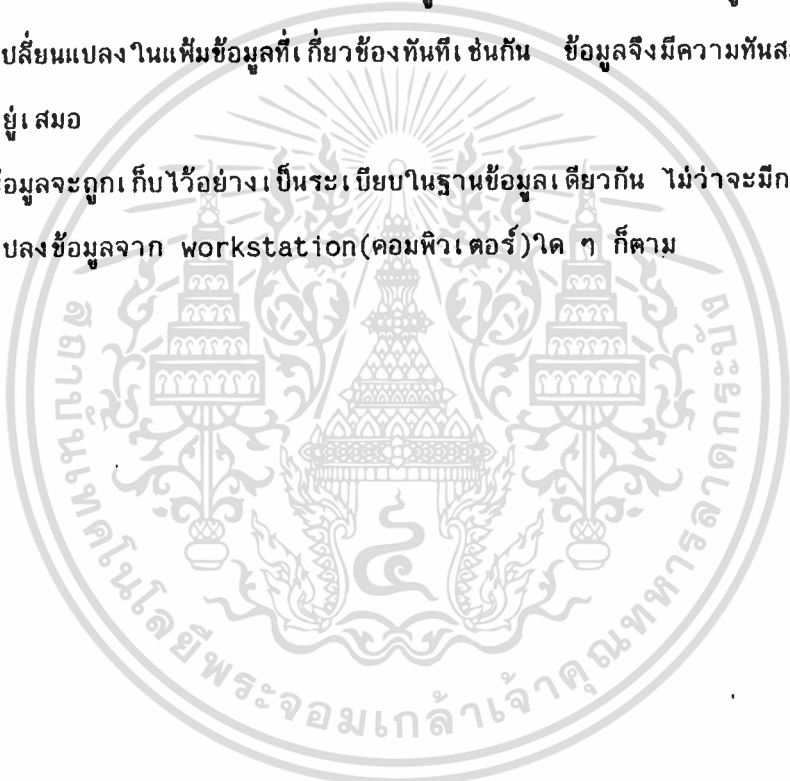
- สามารถนำเข้าข้อมูลในระบบภาษาไทย 25 บรรทัด
- สามารถป้องกันข้อผิดพลาดต่าง ๆ ที่เกิดขึ้นจากการป้อนข้อมูลได้
- สามารถแก้ไข เพิ่มเติม ลบข้อมูลได้ โดยฝ่ายทะเบียนจะเป็นผู้ที่สามารถกระทำการได้ทั้งหมดแต่ถ้าเป็นนักศึกษา หรือ อาจารย์ จะกระทำได้เพียงบางส่วนเท่านั้น
- การป้อนข้อมูลจะนำข้อมูลเข้าทางแป้นพิมพ์และจอภาพทั้งหมด จึงสะดวกในการใช้งาน

เอกสารนี้เป็นเอกสารที่งานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าได้ง่าย
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ลดขั้นตอนในการปฏิบัติงานได้มาก

ส่วนการประมวลผล

- สามารถประมวลผล และค้นหาข้อมูลที่ต้องการได้อย่างรวดเร็ว
- มีรูปแบบการทำงานที่คล้ายคลึงกัน ดังนั้น เมื่อเข้าใจการทำงานอย่างหนึ่งแล้วจะสามารถเข้าใจงานอื่นได้โดยง่าย
- มีการเปลี่ยนแปลงรายละเอียดของข้อมูลทันที เมื่อมีการแก้ไขข้อมูล และจะทำการเปลี่ยนแปลงในแฟ้มข้อมูลที่เกี่ยวข้องทันทีเช่นกัน ข้อมูลจึงมีความทันสมัย และถูกต้องอยู่เสมอ
- ข้อมูลจะถูกเก็บไว้อย่างเป็นระเบียบบนฐานข้อมูลเดียวกัน ไม่ว่าจะมีการแก้ไขเปลี่ยนแปลงข้อมูลจาก workstation(คอมพิวเตอร์)ใด ๆ ก็ตาม



ข้อเสนอแนะ

จากการที่ได้พัฒนาระบบงานทะเบียนให้ใช้กับ LAN และคอมพิวเตอร์มานั้น จะเห็นได้ว่างานบางส่วนยังใช้งานจริงไม่ได้เพราะงานระบบทะเบียนจริงๆ นั้นยังมีรายละเอียดอีกหลายอย่างที่ยังไม่ได้มีการพัฒนามาใช้คอมพิวเตอร์

- การออก Transcript, ใบเกรดและใบลงทะเบียน
- การออก ตารางสอน ตารางสอบ ของนักศึกษาที่ลงทะเบียนแล้ว
- รายละเอียดวิชาต่างๆ ควรจะอยู่ในหน้าจอเดียว ให้สามารถเห็นรายละเอียดได้ที่ละรายวิชา
- ในการลงทะเบียน ควรจะมีการระบุว่าเป็น วิชาบังคับ หรือ วิชาเลือก
- นำฮาร์ดแวร์มาช่วยเสริมให้ระบบการลงทะเบียนสามารถทำได้อย่างรวดเร็ว
- ตรวจสอบ และแสดง รายวิชา จำนวนหน่วยกิต ของวิชาบังคับและวิชาเลือก ของนักศึกษาที่ผ่านรายวิชานั้นๆ แล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* STUDENT */

extern struct var_template stu_var[];
extern struct lbl_template stu_lbl[];

/* File Structre */
extern struct {
    char stu_id[10];
    int part;
    char th_name[21];
    int sex;
    char th_sname[31];
    char ename[21];
    char e_sname[31];
    char d_birth[9];
    char d_cont[9];
    char adv_name[41];
    char add1[41];
    char add2[41];
    char tel[9];
    char p_name[21];
    char p_sname[31];
    char p_add1[41];
    char p_add2[41];
    char p_tel[9];
    int stu_ocrad ;
    float stu_osco ;
    int stu_ncrad ;
    float stu_nsco ;
    int stu_crad ;
    float stu_sco ;
    char stu_stat[10] ;
} stu_rec ;

extern struct {
    char stu_id[10];
    int part;
    char th_name[21];
    int sex;
    char th_sname[31];
    char ename[21];
    char e_sname[31];
    char d_birth[9];
    char d_cont[9];
    char adv_name[41];
    char add1[41];
    char add2[41];
    char tel[9];
    char p_name[21];
    char p_sname[31];
    char p_add1[41];
    char p_add2[41];
    char p_tel[9];
} stu_buf ;
```



```
extern struct {
    char id[10];
} stu_key0;

extern char stu_blk[128];
extern int stu_len ;

extern struct svar_template ssex_var[];
extern struct selv_template sex_selv[];
extern struct var_template sssex_var[];
extern char ssex_buf[5];

extern struct svar_template sdep_var[];
extern struct selv_template dep_selv[];
extern struct var_template ssdep_var[];
extern char sdep_buf[41];

/* Resume Student */
extern struct var_template res_var[] ;
extern struct lbl_template res_lbl[] ;

/* File Structure */
extern struct {
    char stu_id[10];
    char locate1[41];
    char time_int1[10];
    char locate2[41];
    char time_int2[10];
    char locate3[41];
    char time_int3[10];
    char locate4[41];
    char time_int4[10];
} res_rec,res_buf ; /* length 214 */

/* Key Structure */
extern struct {
    char stu_id[10];
} res_key0;

extern char res_blk[128];
extern int res_len ;

/* Total Subject */
extern struct var_template subj_var[] ;
extern struct lbl_template subj_lbl[] ;

/* File Structure */
extern struct {
    char subj_code[7];
    char subj_name[61];
    char subj_ename[61];
    int subj_cred;
    int subj_theory;
    int subj_opr;
    char subj_prev1[7];
```

```

        char subj_prev2[7];
        char subj_prev3[7];
    } subj_rec,subj_buf; /* length 156 */

extern struct {
    char subj_code[7];
    } subj_key0;

extern char subj_blk[128];
extern int  subj_len ;

    /* Open Subject */
extern struct var_template osub_var[] ;
extern struct lbl_template osub_lbl[] ;

    /* File structure */

extern struct {
    char osub_id[7];
    char osub_tea[31];
    int  osub_day ;
    int  osub_part1 ;
    char osub_date[7] ;
    int  osub_part2 ;
    } osub_rec,osub_buf ; /* length 51 */

extern struct {
    char osub_id[7];
    }osub_key0;

extern char osub_blk[128];
extern int  osub_len ;

    /* Part Selection */
extern struct svar_template spa_var[] ;
extern struct selv_template pa_selv[] ;
extern struct var_template sspal_var[];
extern struct var_template sspa2_var[];
extern char  spa1_buf[5];
extern char  spa2_buf[5];

extern struct svar_template sdate_var [] ;
extern struct selv_template date_selv[] ;
extern struct var_template ssdate_var[] ;
extern char  sdate_buf[9] ;

    /* General Subjects */
extern struct var_template gens_var[] ;
extern struct lbl_template gens_lbl[] ;

    /* File Structre */
extern struct {
    int  gens_dep ;
    int  gens_year ;
    int  gens_sem ;

```

```

        char code[7] ;
    } gens_rec ,gens_buf ; /* length 13 */

extern struct {
    int    gens_dep ;
    int    gens_year ;
    int    gens_sem ;
    char   code[7] ;
    } gens_key0;

extern char gens_blk[128];
extern int  gens_len;

extern struct var_template    gsdep_var[];
extern struct var_template    gssem_var[];
extern char gsem_buf[7];
extern char gsdep_buf[41] ;

/* Select Subjects */
extern struct var_template selct_var[] ;
extern struct lbl_template selct_lbl[] ;

/* File Structre */
extern struct {
    int    selct_dep ;
    int    selct_sem ;
    int    ksub ;
    char   slcode[7] ;
    } selct_rec ,selct_buf ; /* length 52 */

extern struct {
    int    selct_dep ;
    int    selct_sem ;
    int    ksub ;
    char   slcode[7] ;
    } selct_key0;

extern char selct_blk[128];
extern int  selct_len ;

extern struct var_template sldep_var[] ;
extern struct svar_template ssem_var[];
extern struct selv_template ssem_selv[];
extern struct var_template slsem_var[] ;
extern struct svar_template slsub_var[] ;
extern struct selv_template slsub_selv[] ;
extern struct var_template ksub_var[] ;
extern char slsem_buf[7];
extern char sldep_buf[41];
extern char slsub_buf[21];

/* Plan Education */
extern struct var_template plan_var[] ;
extern struct lbl_template plan_lbl[] ;

```

```

/* File Structure */
extern struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
    struct {
        char    code[7] ;
    } sub[12] ;
} plan_buf ;

extern struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
    struct {
        char    code[7] ;
    } sub[12] ;
} plan_rec ; /* length 90 */

/* Key Structure */
extern struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
} plan_key0 ;

extern char    plan_blk[128] ;
extern int     plan_len ;

extern struct var_template    pldep_var[] ;
extern char    pldep_buf[41] ;
extern struct var_template    plsem_var[] ;
extern char    plsem_buf[7] ;

/* Interval of Regising */
extern struct var_template int_var[];
extern struct lbl_template int_lbl[];

/* File Structure Interval Regising */
extern struct {
    char    st1[9];
    char    endl[9];
    char    st2[9];
    char    end2[9];
}int_rec,int_buf;/* length 36 */

extern struct {
    char    st1[9];
} int_key0;

extern char int_blk[128];
extern int  int_len;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* Change Grade For Regist Grade */
extern struct var_template cgr_var[];
extern struct lbl_template cgr_lbl[];

/* File Structure */

extern struct {
    char cgr_sub[7];
    char code[10];
    char grade[3];
} cgr_rec,cgr_buf ; /* length 20 */

/* Key Structure */
extern struct {
    char cgr_sub[7];
    char code[10];
} cgr_key0;

extern char cgr_blk[128];
extern int cgr_len ;

/* Mon */
extern struct var_template mon_var[] ;
extern struct lbl_template mon_lbl[] ;

extern struct {
    char id[10] ;
    int depp;
    int sem ;
    int year ;
    struct {
        char code[7] ;
    } sub[12] ;
} mon_rec,mon_buf ;/* length 100 */

/* Key structure */

extern struct {
    char id[10] ;
    int depp;
    int sem;
    int year;
} mon_key0 ;

extern char mon_blk[128] ;
extern int mon_len ;

extern struct var_template mdep_var[];

extern char mdep_buf[41] ;
extern struct var_template mplsem_var[];
extern char msem_buf[7] ;

/* Input Grade */
extern struct var_template inp_var[];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดก็ตามถ้ามีการแก้ไขให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

extern struct lbl_template    inp_lbl[];

extern struct {
    char    id[10] ;
    int     depp;
    int     sem ;
    int     year ;
    struct {
        char    code[7] ;
        char    grd[3] ;
    } sub[12] ;
    int     oldtot ;
    float    oldsco ;
    int     newtot ;
    float    newsco ;
    int     total ;
    float    sco ;
    float    gps ;
} inp_rec, inp_buf ;    /* length 148 */

extern struct {
    char    id[10] ;
    int     depp;
    int     sem ;
    int     year ;
} inp_key0 ;

extern char    inp_blk[128] ;
extern int     inp_len ;
extern struct var_template    inpdep_var[] ;

extern struct svar_template    ipsem_var[];
extern struct selv_template    ipsem_selv[];
extern struct var_template    inpsem_var[] ;

extern char    inpsem_buf[7] ;
extern char    inpdep_buf[41] ;

/* Teacher Input Grade */
extern struct var_template    tgr_var[];
extern struct lbl_template    tgr_lbl[];

/* File Structure */
extern struct {
    char tgr_sub[7];
    char code[10];
    char gradel[3] ;
    char grade2[3] ;
} tgr_rec,tgr_buf ; /* length 24 */

/* Key Structure */
extern struct {
    char tgr_sub[7] ;
    char code[10] ;
} tgr_key0;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามแก้ไขเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
extern tgr_blk[128];
extern int tgr_len;

/* Conclude Semester */
extern struct var_template csem_var[] ;
extern struct lbl_template csem_lbl[] ;

/* File Structure */
extern struct {
    char    code[10] ;
    int     dep ;
    int     sem ;
    int     year ;
    int     ocra ;
    int     scra ;
    int     cra ;
    float   gps ;
    float   gpa ;
    float   ngpa ;
} csem_rec , csem_buf ; /* length 34 */

extern struct {
    char    code[10] ;
    int     dep ;
    int     sem ;
    int     year ;
} csem_key0 ;

extern char    csem_blk[128] ;
extern int     csem_len ;
extern struct var_template csemdep_var[] ;
extern struct var_template csemsem_var[] ;
extern char    csemdep_buf[41] ;
extern char    csemsem_buf[7] ;

/* RESUME FOR STUDENT */

extern struct var_template st_var[];

extern struct lbl_template st_lbl[] ;
```

```

/* STUDENT */
struct var_template stu_var[] = {
/* CODE */      {3,20,STRING,"99-999999","99-999999"},
/* DEPART */    {4,20,INT,"9","9"},
/* NAME 20 */   {5,20,STRING,"XXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXX"},
/* SEX */       {5,55,INT,"9","9"},
/* SUR NAME */  {6,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"},
/* E_NAME */    {7,20,STRING,"XXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXX"},
/* ENG S_NAME */ {8,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"},
/* DATE BIRTH */ {9,20,STRING,"99/99/99","99/99/99"},
/* DATE CONT */ {10,20,STRING,"99/99/99","99/99/99"},
/* ADVISOR NAME */ {11,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X"},
/* ADDRESS 1*/  {12,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X"},
/* ADDRESS 2*/  {13,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X"},
/* TELEPHONE */ {14,20,STRING,"999-9999","999-9999"},
/* P_NAME */    {15,20,STRING,"XXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXX"},
/* P_SUR NAME */ {16,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX"},
/* P_ADDRESS 1*/ {17,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X"},
/* P_ADDRESS 2*/ {18,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X",
                  "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
X"},
/* P_TELEPHONE */ {19,20,STRING,"999-9999","999-9999"},
};

```

```

struct lbl_template stu_lbl[] = {
{3,9,"รหัสนักศึกษา:",NM_VDO},
{4,12,"ภาควิชา:",NM_VDO},
{5,11,"ชื่อ(ไทย):",NM_VDO},
{5,50,"เพศ:",NM_VDO},
{6,7,"นามสกุล(ไทย):",NM_VDO},
{7,14,"NAME:",NM_VDO},
{8,11,"SURNAME:",NM_VDO},
{9,13,"นามสกุล:",NM_VDO},
{10,4,"วันเข้การศีกษา:",NM_VDO},
{11,10,"อ.ทศ.ป.ร.ษา:",NM_VDO},
{12,10,"ทศ.ยง.ป.จ.จ.บ.น:",NM_VDO},

```

```

        {12,10,"ทศ.เกษตร.ป.จ.จ.บ.ม.",NM_VDO},
        {14,12,"โทรศ.พท.ร.",NM_VDO},
        {15,9,"ศ.เกษตร.ป.จ.จ.บ.ม.",NM_VDO},
        {16,12,"นามสกุล.",NM_VDO},
        {17,10,"ทศ.เกษตร.ป.จ.จ.บ.ม.",NM_VDO},
        {17,10,"ทศ.เกษตร.ป.จ.จ.บ.ม.",NM_VDO},
        {19,12,"โทรศ.พท.ร.",NM_VDO}
    };

```

```
/* File Structure */
```

```

struct {
    char    stu_id[10];
    int     part;
    char    th_name[21];
    int     sex;
    char    th_sname[31];
    char    ename[21];
    char    e_sname[31];
    char    d_birth[9];
    char    d_cont[9];
    char    adv_name[41];
    char    add1[41];
    char    add2[41];
    char    tel[9];
    char    p_name[21];
    char    p_sname[31];
    char    p_add1[41];
    char    p_add2[41];
    char    p_tel[9];
    int     stu_ocrad ;
    float   stu_osco ;
    int     stu_ncrad ;
    float   stu_nsco ;
    int     stu_crad ;
    float   stu_sco ;
    char    stu_stat[10] ;
} stu_rec ;

```

```

struct {
    char    stu_id[10];
    int     part;
    char    th_name[21];
    int     sex;
    char    th_sname[31];
    char    ename[21];
    char    e_sname[31];
    char    d_birth[9];
    char    d_cont[9];
    char    adv_name[41];
    char    add1[41];
    char    add2[41];
    char    tel[9];
    char    p_name[21];
    char    p_sname[31];
    char    p_add1[41];
    char    p_add2[41];

```

```

char p_tel[9];
} stu_buf ;/* 411 */

/* Key Structure */
struct {
    char id[10] ;
    } stu_key0;

char stu_blk[128];
int stu_len = sizeof(stu_rec) ;

/* Part Selection */
struct svar_template ssex_var[] = {
    {"ชาย",NULL,NULL},
    {"หญิง",NULL,NULL}
};

struct selv_template sex_selv[] = {{2,0,0}};
struct var_template sssex_var[]={5,55,STRING,"XXX","XXX",}};
char ssex_buf[5];

struct svar_template sdep_var [] = {
    {"คณะศิลปกรรมและวิทยาการคอมพิวเตอร์",NULL,NULL},
    {"คณะอุตสาหกรรม",NULL,NULL},
    {"เทคโนโลยีชีวภาพ",NULL,NULL},
    {"พลังศึกษาระยะภาค",NULL,NULL},
    {"สังคมศึกษาระยะภาค",NULL,NULL}
};

struct selv_template dep_selv[] = {{5,0,0}};
struct var_template ssdep_var[] = {{4,20,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX",}};
char sdep_buf[41];

/* STUDENT RESUME */
struct var_template res_var[] = {
/* Student Code */      {4,16,STRING,"99-999999","99-999999"},
/* Sec_name */          {9,24,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XX",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX"},
/* Time Interval */     {9,69,STRING,"9999-9999","9999-9999"},
/* Sec_name */          {10,24,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX"},
/* Time Interval */     {10,69,STRING,"9999-9999","9999-9999"},
/* Sec_name */          {11,24,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX"},
/* Time Interval */     {11,69,STRING,"9999-9999","9999-9999"},
/* Sec_name */          {12,24,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXX",
"XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX"}
};

```

```
XXX"},},  
/* Time Interval */      {12,69,String,"9999-9999","9999-9999"},  
                          };  
  
struct lbl_template res_lbl[] = {  
    {4,4,"รท.สน.ภคปกษา :",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO},  
    {2,2," ",NM_VDO}  
};  
  
/* File Structure */  
struct {  
    char stu_id[10];  
    char locatel[41];  
    char time_intl[10];  
    char locate2[41];  
    char time_int2[10];  
    char locate3[41];  
    char time_int3[10];  
    char locate4[41];  
    char time_int4[10];  
} res_rec,res_buf ; /* length 214 */  
  
/* Key Structure */  
struct {  
    char stu_id[10];  
} res_key0;  
  
char res_blk[128];  
int res_len = sizeof(res_rec);  
  
/* Total Subject */  
struct var_template subj_var[] = {  
    /* Subject Code */     {5,20,String,"99-999","99-999"},,  
    /* Subject Name */     {6,20,String,"xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
xxxxxxxxxxxxxxxxxxxxxxxxx",  
                                "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
xxxxxxxxxxxxxxxxxxxxxxxxx"},,  
    /* Eng_name */         {7,20,String,"xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
xxxxxxxxxxxxxxxxxxxxxxxxx",  
                                "xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  
xxxxxxxxxxxxxxxxxxxxxxxxx"},,  
    /* Credit */           {8,20,INT,"9","9"},,  
    /* Theorem */          {9,20,INT,"9","9"},,  
    /* Operate */          {10,20,INT,"9","9"},,  
    /* Previous */         {11,20,String,"99-999","99-999"},,  
    /* Extra */            {11,29,String,"99-999","99-999"},,  
    /* Note */             {11,38,String,"99-999","99-999"},}
```

```

struct lbl_template subj_lbl[] = {
    {5,12,"รหัสวิชา:",NM_VDO},
    {6,13,"ชื่อวิชา:",NM_VDO},
    {7,6,"ชื่อวิชา(อังกฤษ):",NM_VDO},
    {8,12,"หน่วยกิต:",NM_VDO},
    {9,14,"ทฤษฎี:",NM_VDO},
    {10,14,"ปฏิบัติ:",NM_VDO},
    {11,9,"วิชาเอก/อนโทนา:",NM_VDO},
    {11,9,"วิชาเอก/อนโทนา:",NM_VDO},
    {11,9,"วิชาเอก/อนโทนา:",NM_VDO}
};

/* File Structure */
struct {
    char subj_code[7];
    char subj_name[61];
    char subj_ename[61];
    int subj_cred;
    int subj_theory;
    int subj_opr;
    char subj_prev1[7];
    char subj_prev2[7];
    char subj_prev3[7];
} subj_rec,subj_buf; /* length 156 */

/* Key Structure */
struct {
    char subj_code[7];
} subj_key0;

char subj_blk[128];
int subj_len = sizeof(subj_rec);

/* Open Subject */
struct var_template osub_var[] = {
    /* Sub Code */      {11,22,STRING,"99-999","99-999",},
    /* Name_Teacher */ {14,22,STRING,"XXXXXXXXXXXXXXXXXXXXXXXXXXXX",
                                "XXXXXXXXXXXXXXXXXXXXXXXXXXXX",},
    /* Study Day */     {15,25,INT,"9","9",},
    /* Interval */      {15,58,INT,"9","9",},
    /* Date Test */     {16,22,STRING,"99/99","99/99",},
    /* Interval */      {16,58,INT,"9","9",}
};

struct lbl_template osub_lbl[] = {
    {11,13,"รหัสวิชา :",NM_VDO},
    {14,9,"อาจารย์ผะสอน :",NM_VDO},
    {15,14,"วนสอน :",NM_VDO},
    {15,47,"เวลาเรียน :",NM_VDO},
    {16,14,"วนสอบ :",NM_VDO},
    {16,52,"ช.ว.ง :",NM_VDO}
};

```



```

char osub_id[7] ;
char osub_tea[31] ;
int osub_day ;
int osub_part1 ;
char osub_date[7] ;
int osub_part2 ;
    } osub_rec,osub_buf; /* length 51 */

struct {
    char osub_id[7];
    }osub_key0;

char osub_blk[128];
int osub_len = sizeof(osub_rec);

/* Part Selection */
struct svar_template spa_var[] = {
    {"เช้า",NULL,NULL},
    {"บ่าย",NULL,NULL}
};

struct selv_template pa_selv[] = { { 2,0,0, } };
struct var_template sspal_var[]={ {15,58,STRING,"xxx","xxx",}};
struct var_template sspa2_var[]={ {16,58,STRING,"xxx","xxx",}};
char spa1_buf[5];
char spa2_buf[5];
char trn[128];
int trn_len;

struct svar_template sdate_var [] = {
    {"จันทร์",NULL,NULL},
    {"อังคาร",NULL,NULL},
    {"พุธ",NULL,NULL},
    {"พฤหัสบดี",NULL,NULL},
    {"ศุกร์",NULL,NULL},
    {"เสาร์",NULL,NULL},
    {"อาทิตย์",NULL,NULL}
};

struct selv_template date_selv[] = {{7,0,0}};
struct var_template ssdate_var[] = {{15,25,STRING,"XXXXXXXX","XXXXXXXX",}};

char sdate_buf[9];

/* General Subjects */
struct var_template gens_var[]= {
/* DEPARTMENT */    { 4,20,INT,"9","9",},
/* YEAR */          { 5,20,INT,"9","9",},
/* SEMESTER */      { 5,41,INT,"9","9",},
/* ID-SUBJECTS */   { 7,20,STRING,"99-999","99-999",},
};

/*
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ
*/
struct lbl_template gens_lbl[] = { และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
    {4,11,"ภาค๖ : ",NM_VDO},

```

```
{5,13,"ชนิดของท่อ : ",NM_VDO},
{5,27,"ภาคระบอบท่อ : ",NM_VDO},
{7,11,"รหัสสาขา : ",NM_VDO},
};
```

```
/* File Structre */
```

```

struct {
    int    gens_dep ;
    int    gens_year ;
    int    gens_sem ;
    char   code[7] ;
} gens_rec ,gens_buf ; /* length 13 */

```

```
struct {
```

```

    int    gens_dep ;
    int    gens_year ;
    int    gens_sex ;
    char   code[7] ;
} gens_key0;

```

```
char gens_blk[128];
int gens_len = sizeof(gens_rec);
```

[illegible]

```
struct var_template gssem_var[]={5,41,STRING,"XXXXXX","XXXXXX",}};
```

```
char gsem_buf[7];
char gsdep_buf[41];
```

```
/* Select Subjects */
```

```
struct var_template select_var[] = {
/* DEPARTMENT */      {5,12,INT,"9","9"},
/* SEMESTER */        {5,64,INT,"9","9"},
/* KINDS */           {8,20,INT,"9","9"},
/* ID-SUBJECTS */     {9,20,STRING,"99-999","99-999"},
};
```

```

struct lbl_template select_lbl[] = {
    {5,3,"ภาควิชา : ",NM_VDO},
    {5,53,"ภาคเรียนที่ : ",NM_VDO},
    {8,10,"วิชาเลือก : ",NM_VDO},
    {9,11,"รหัสวิชา : ",NM_VDO}
};

```

```
/* File Structre */
```

```
struct {
    int selct_dep ;
    int selct_sem ;
    int ksub ;
    char slcode[7] ;
}
```

int selct_dep ;

```
int selct_sem ;
```

int ksub ; นี้อา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
char  slcode[7] ;
```

```

} select_rec ,select_buf ; /* length 13 */

struct {
    int    select_dep ;
    int    select_sem ;
    int    ksub ;
    char    slcode[7] ;
} select_key0;

char    select_blk[128];
int    select_len = sizeof(select_rec);

struct var_template    sldep_var[] = {{5,12,STRING,"XXXXXXXXXXXXXXXXXXXXX",
XXXXXXXXXXXXXXXXXXXXX",
XXXXXXXXXXXXXXXXXXXXX",}};
struct svar_template    ssem_var[] = {
    {"1st",NULL,NULL},
    {"2nd",NULL,NULL},
    {"Summer",NULL,NULL}
    };
struct selv_template    ssem_selv[] = {{3,0,0}};
struct var_template    slsem_var[] = {{5,64,STRING,"XXXXXX",
XXXXXX",}};
struct svar_template    slsub_var[] = {
    {"กล มวิชาสังคมศาสตร์",NULL,NULL},
    {"กล มวิชามนุษยศาสตร์",NULL,NULL},
    {"กล มวิชาคณิตศาสตร์",NULL,NULL},
    {"กล มวิชาคอมพิวเตอร์",NULL,NULL},
    {"กล มวิชาชีววิทยา",NULL,NULL},
    {"กล มวิชาฟิสิกส์",NULL,NULL},
    {"กล มวิชาสถิติ",NULL,NULL},
    {"กล มวิชาคอมพิวเตอร์",NULL,NULL},
    {"กล มวิชาเลอิกอัส",NULL,NULL},
    {"กล มวิชาวิทยาศาสตร์",NULL,NULL},
    {"กล มวิชาภาษา",NULL,NULL}
    };
struct selv_template    slsub_selv[] = {{11,0,0}};
struct var_template    ksub_var[] = {{8,20,STRING,"XXXXXXXXXXXXXXXXXXXXX",
XXXXXXXXXXXXXXXXXXXXX",}};

char    slsem_buf[7];
char    sldep_buf[41];
char    slsub_buf[21];

/* Plan Education */
struct var_template    plan_var[] = {
/* Department */    { 4,17,INT,"9","9"},
/* Semester */    { 5,17,INT,"9","9"},
/* Year */    { 5,34,INT,"9","9"},
    { 7, 3,STRING,"99-999","99-999"},
    { 8, 3,STRING,"99-999","99-999"},

```

```
{ 9, 3, STRING, "99-999", "99-999", },
{10, 3, STRING, "99-999", "99-999", },
{11, 3, STRING, "99-999", "99-999", },
{12, 3, STRING, "99-999", "99-999", },
{13, 3, STRING, "99-999", "99-999", },
{14, 3, STRING, "99-999", "99-999", },
{15, 3, STRING, "99-999", "99-999", },
{16, 3, STRING, "99-999", "99-999", },
{17, 3, STRING, "99-999", "99-999", },
{18, 3, STRING, "99-999", "99-999", }
};
```

[illegible]

```

struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
    struct {
        char    code[7] ;
    } sub[12] ;
} plan buf ;

```

```

struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
    struct {
        char    code[7] ;
    } sub[12] ;
} plan_rec ; /* length 90 */

```

```

struct {
    int    plan_dep ;
    int    plan_sem ;
    int    plan_year ;
} plan_key0 ;
char    plan_blk[I28] ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
int plan_year ;
} plan_key0 ;
```

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int    plan_len = sizeof(plan_rec) ;

struct var_template  pldep_var[] = { { 4,17,STRING,"xxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxx",
                                     "xxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxx",} } ;
char    pldep_buf[41] ;
struct var_template  plsem_var[] = { { 5,17,STRING,"xxxxxx",
                                     "xxxxxx",} } ;

char    plsem_buf[7] ;

/* Interval of Regising */
struct var_template int_var[] = {
/* Start 1 */      {7,20,STRING,"99/99/99","99/99/99"},
/* End 1   */      {8,20,STRING,"99/99/99","99/99/99"},
/* Start 2 */      {12,20,STRING,"99/99/99","99/99/99"},
/* End 2   */      {13,20,STRING,"99/99/99","99/99/99"},
                    };
struct lbl_template int_lbl[] = {
                    {7,13,"เร.มต.น:",NM_VDO},
                    {8,14,"ส.นส.ด:",NM_VDO},
                    {12,13,"เร.มต.น:",NM_VDO},
                    {13,14,"ส.นส.ด:",NM_VDO}
                    };

/* File Structure Interval Regising */
struct {
    char st1[9];
    char end1[9];
    char st2[9];
    char end2[9];
}int_rec,int_buf; /* length 36 */

struct {
    char st1[9];
} int_key0;

char int_blk[128];
int  int_len = sizeof(int_rec);

/* Chage Grade For Register Grade */
struct var_template  cgr_var[] = {
/* Subj code   */      {4,20,STRING,"99-999","99-999"},
/* Studen Code */      {8,20,STRING,"99-999999","99-999999"},
/* New Grade   */      {11,64,STRING,"XX","XX"},
                    };

struct lbl_template  cgr_lbl[] = {
                    {4,11,"รหัสวิชา :",NM_VDO},
                    {8,8,"รหัสนศ.ศึกษา :",NM_VDO},
                    {11,52,"เกรดป.จ.น. :",NM_VDO}
                    };

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามบิดพลิ้วแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/* File Structure */

```

```

struct {
    char    cgr_sub[7];
    char    code[10];
    char    grade[3] ;
    } cgr_rec,cgr_buf ; /* length 20 */

/* Key Structure */
struct {
    char    cgr_sub[7];
    char    code[10];
    } cgr_key0;

char    cgr_blk[128];
int    cgr_len = sizeof (cgr_rec);

/* Mon */
struct var_template    mon_var[] = {
/* Identity student */ { 3,18,STRING,"99-999999","99-999999"},
/* Department */      { 5,18,INT,"9","9"},
/* Semester */        { 6,18,INT,"9","9"},
/* Year */             { 6,34,INT,"9","9"},
    } ;

struct lbl_template mon_lbl[] = {
    { 3, 6,"รหัส นศ.ศึกษา :",NM_VDO},
    { 5, 9, "ภาควิชา :",NM_VDO},
    { 6, 7,"ภาคเรียนที่ :",NM_VDO},
    { 6,27,"ชั้นปีที่ :",NM_VDO}
    } ;

struct {
    char    id[10] ;
    int     depp;
    int     sem ;
    int     year ;
    struct {
        char    code[7] ;
    } sub[12] ;
    } mon_rec,mon_buf ; /* length 100 */

struct {
    char    id[10] ;
    int     depp;
    int     sem;
    int     year;
    } mon_key0 ;

char    mon_blk[128] ;
int    mon_len = sizeof (mon_rec) ;

struct var_template    mdep_var[] = { { 5,18,STRING,"xxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxx",

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำซ้ำโดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[illegible]

```

/* Input Grade */
struct var_template    inp_var[] = {

/* Identity student */ { 3,18,STRING,"99-999999","99-999999"},
/* Department */      { 5,18,INT,"9","9"},
/* Semester */        { 6,18,INT,"9","9"},
/* Year */             { 6,34,INT,"9","9"},
/* Grade */           { 8,70,STRING,"XX","XX"},
                      { 9,70,STRING,"XX","XX"},
                      {10,70,STRING,"XX","XX"},
                      {11,70,STRING,"XX","XX"},
                      {12,70,STRING,"XX","XX"},
                      {13,70,STRING,"XX","XX"},
                      {14,70,STRING,"XX","XX"},
                      {15,70,STRING,"XX","XX"},
                      {16,70,STRING,"XX","XX"},
                      {17,70,STRING,"XX","XX"},
                      {18,70,STRING,"XX","XX"},
                      {19,70,STRING,"XX","XX"},
                      } ;

```

[illegible]

```
/* File Structure */
struct {
```

```
char    id[10] ;
int     depp;
int     sem ;
int     year ;
struct {
    char    code[7] ;
```

```

        char    grd[3] ;
    } sub[12] ;
    int    oldtot ;
    float  oldsco ;
    int    newtot ;
    float  newsco ;
    int    total ;
    float  sco ;
    int    gps ;
} inp_rec,inp_buf ;    /* length 154 */

struct {
    char    id[10] ;
    int     depp;
    int     sem ;
    int     year ;
} inp_key0 ;

char    inp_blk[128] ;
int     inp_len = sizeof(inp_rec);

struct var_template    inpdep_var[] = { { 5,18,STRING,"xxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxx",
xxxxxxxxxxxxxxxxxxxxxxxxxxxx",}} ] ;

struct svar_template    ipsem_var[] = {
    {"1st",NULL,NULL},
    {"2nd",NULL,NULL},
    {"Summer",NULL,NULL}
    };

struct selv_template    ipsem_selv[] = {{3,0,0}};
struct var_template    inpsem_var[] = {{6,18,STRING,"XXXXXX","XXXXXX",}} ;

char    inpdep_buf[41] ;
char    inpsem_buf[7] ;

/* Teacher Input Grade */
struct var_template    tgr_var[] ={
/* Subj code    */    {4,25,STRING,"99-999","99-999",},
/* Studen Code */    {8,25,STRING,"99-999999","99-999999",},
/* Grade      */    {11,25,STRING,"XX","XX",},
/* Grade      */    {12,25,STRING,"XX","XX",}
    };

struct lbl_template    tgr_lbl[] = {
    {4,16,"รหัสวิชา :",NM_VDO},
    {8,13,"รหัสแผนกศึกษา :",NM_VDO},
    {11,18,"เกรด :",NM_VDO},
    {12,13,"เกรดภาคเรียน :",NM_VDO}
    };

/* File Structure */
struct {
    char    tgr_sub[7];

```

```

        char    code[10];
        char    grade1[3] ;
        char    grade2[3] ;
    } tgr_rec,tgr_buf ; /* length 23 */

/* Key Structure */
struct {
        char    tgr_sub[7];
        char    code[10];
    } tgr_key0;

char    tgr_blk[128];
int     tgr_len = sizeof (tgr_rec);

/* Conclude Semester */
struct var_template    csem_var[] = {
/* ID-Student */      {7,22,STRING,"99-9999999","99-9999999"},
/* Department */      {9,22,INT,"9","9"},
/* Semester */        {10,22,INT,"9","9"},
/* Year inp */         {10,45,INT,"9","9"},
                        };

struct lbl_template    csem_lbl[] = {
                        {7, 8,"รหัสนักศึกษา :",NM_VDO},
                        {9,11,"ภาควิชา :",NM_VDO},
                        {10,9,"ภาคเรียนที่ :",NM_VDO},
                        {10,37,"ชั้นปี :",NM_VDO}
                        };

/* File Structure */
struct {
        char    code[10] ;
        int     dep ;
        int     sem ;
        int     year ;
        int     ocra ;
        int     scra ;
        int     cra ;
        float    gps ;
        float    gpa ;
        float    ngpa ;
    } csem_rec , csem_buf ; /* length 34 */

struct {
        char    code[10] ;
        int     dep ;
        int     sem ;
        int     year ;
    } csem_key0 ;

char    csem_blk[128] ;
int     csem_len = sizeof(csem_rec) ;

struct var_template    csemdep_var[] = {{9,22,STRING,"xxxxxxxxxxxxxxxxxxxx

```

```

                                                                 "xxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx",)}} ;

struct var_template    csemsem_var[] = {{10,22,STRING,"XXXXXX","XXXXXX",}}
;

char    csemdep_buf[41] ;
char    csemsem_buf[5] ;

/* RESUME FOR STUDENT */
struct var_template st_var[] = {
    {15,40,STRING,"99-999999","99-999999",}
};

struct lbl_template st_lbl[] = {
    {15,29,"รท.สน.ภค.ภษา :",NM_VDO}
};
```



```

#include "global.inc"
#include "stuu.inc"
main ()
{
    int func_no;
    clrscr();
    set_crt(2);
    outp(0x3b8,0x28);
    outp(0x3ba,0x80);
    outp(0x3bf,0x03);
    cur_off();
    if (open_file() != 0)
    {
        bt_error();
        clrscr();
        cur_on();
        set_crt(0);
        outp(0x3ba,0x0);
        outp(0x3bf,0);
        exit(0);
    }
    header();
    assign_ptr();
    do
    {
        switch (func_no = menu0())
        {
            case F1:
                sys_stud();
                break;
            case F2:
                note_sub();
                break;
            case F3:
                regis_input();
                break;
            case F4:
                result_edu();
                break;
        }
    } while (func_no != F9 );
    if (close_file() != 0)
        bt_error();
    clrscr();
    cur_on();
    set_crt(0);
    outp(0x3ba,0x0);
    outp(0x3bf,0);
}

menu0()
{
    static int func_no = F1;
    static char *m_item[] = {
        "1. ประวัตินักศึกษา",
        "2. การบวชการวายวช",
        "3. การลงทะเบียนนักศึกษา",
        "4. ผลการศึกษาของนักศึกษา",
    }
}

```

```

"9. ออกจากระบบ"    });

func_no = menu_func("ระบบทะเบียนนักศึกษา(ฝ่ายทะเบียน)",F4,m_item,func_no);
return(func_no);
}
sys_stud ()
{
    int func_no ;
    do
    {
        switch (func_no = menu1()).
        {
            case F1:
                student ();
                break;
            case F2:
                resume_edu ();
                break;
        }
    } while(func_no !=F9);
}
menu1()
{
    static int func_no =F1;
    static char *m_nul[] = {
        "1. การบันทึกประวัติส่วนตัว",
        "2. การบันทึกประวัติการศึกษา",
        "9. ออกส่ง,รายการหลัก"
    };
    func_no = menu_func ("ประวัตินักศึกษา",F2,m_nul,func_no) ;
    return (func_no);
}
note_sub()
{
    int func_no ;
    do {
        switch (func_no = menu2())
        {
            case F1:
                gen_sub();
                break;
            case F2:
                selct_sub();
                break;
            case F3:
                open_sub ();
                break;
            case F4:
                list_sub();
                break;
            case F5:
                plan_edu () ;
                break;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

    } while(func_no !=F9);
}
menu2()
{
    static int func_no =F1;
    static char *m_nu2[] = {
        "1. หมวดวิชาบังคับเฉพาะภาคการศึกษา",
        "2. หมวดวิชาเลือกเฉพาะภาคการศึกษา",
        "3. รายละเอียดตัววิชาที่เรียนสอนปัจจุบัน",
        "4. รายละเอียดตัววิชาทั้งหมด",
        "5. โครงสร้างหลักสูตร",
        "9. ออกลิส,รายการหลัก"
    };

    func_no = menu_func ("การปฐมนิเทศวิชา",F5,m_nu2,func_no);
    return (func_no);
}
regis_input()
{
    int func_no ;
    do
    {
        switch (func_no = menu3())
        {
            case F1:
                interval_reg();
                break;
            case F2:
                conclude_reg();
                break;
        }
    } while(func_no !=F9);
}
menu3()
{
    static int func_no =F1;
    static char *m_nu3[] = {
        "1. กำหนดช.เว.ว.นลงทะเบ.ย",
        "2. สร.ปผลการลงทะเบ.ย",
        "9. ออกลิส,รายการหลัก"
    };

    func_no = menu_func ("การลงทะเบียนน.ศ.ศึกษา",F2,m_nu3,func_no);
    return (func_no);
}
result_edu()
{
    int func_no ;
    do {
        switch (func_no = menu4())
        {
            case F1:
                input_grade ();
                break;
            case F2:
                change_grade();
                break;
        }
    }
}

```

```

        case F3:
            conclude_semester() ;
            break;
        case F4:
            /* conclude_total(); */
            break;
    }
} while(func_no !=F9);
}
menu4()
{
    static int func_no =F1;
    static char *m_nu4[] = {
        "1. การบันทึกเกรด",
        "2. การแก้ไขเกรด",
        "3. สรุปผลเฉพาะภาคการศึกษา",
        "4. สรุปผลทั้งหมดท.ผ.วนมา",
        "9. ออกสข.รายการผลน"
    };

    func_no = menu_func ("ผลการศษษาของนักเรียน",F4,m_nu4,func_no);
    return (func_no);
}
open_file()
{
    STATUS = BTRV (B_OPEN,stu_blk,&stu_rec,&stu_len,"STU1.BTR",1);
    if(STATUS !=0)
        return(STATUS);
    STATUS = BTRV (B_OPEN,res_blk,&res_rec,&res_len,"RESUME1.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,subj_blk,&subj_rec,&subj_len,"TSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,osub_blk,&osub_rec,&osub_len,"OSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
    if(STATUS !=0)
        return(STATUS);
    STATUS = BTRV (B_OPEN,int_blk,&int_rec,&int_len,"INTER.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,gens_blk,&gens_rec,&gens_len,"GENSUB.BTR",1);
    if(STATUS !=0)
        return (STATUS) ;
    STATUS = BTRV (B_OPEN,selct_blk,&selct_rec,&selct_len,"SELECT.BTR",
1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,tgr_blk,&tgr_rec,&tgr_len,"TGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,cgr_blk,&cgr_rec,&cgr_len,"CGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เฉพาะเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

STATUS = BTRV (B_OPEN,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,csem_blk,&csem_rec,&csem_len,"CSEM.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
return(0);
}
close_file()
{
    STATUS = BTRV (B_CLOSE,stu_blk,&stu_rec,&stu_len,"STU1.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,res_blk,&res_rec,&res_len,"RESUME1.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,subj_blk,&subj_rec,&subj_len,"TSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,osub_blk,&osub_rec,&osub_len,"OSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,int_blk,&int_rec,&int_len,"INTER.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,gens_blk,&gens_rec,&gens_len,"GENSUB.BTR",1)
;
    if(STATUS !=0)
        return (STATUS) ;
    STATUS = BTRV (B_CLOSE,selct_blk,&selct_rec,&selct_len,"SELECT.BTR"
,1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,tgr_blk,&tgr_rec,&tgr_len,"TGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,cgr_blk,&cgr_rec,&cgr_len,"CGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,csem_blk,&csem_rec,&csem_len,"CSEM.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    return(0);
}

```

```

}
assign_ptr()
{
    int i ;

    /* Student File */
    stu_var[0].datatag.str_add = stu_buf.stu_id;
    stu_var[1].datatag.int_add = &stu_buf.part;
    stu_var[2].datatag.str_add = stu_buf.th_name;
    stu_var[3].datatag.int_add = &stu_buf.sex;
    stu_var[4].datatag.str_add = stu_buf.th_sname;
    stu_var[5].datatag.str_add = stu_buf.ename;
    stu_var[6].datatag.str_add = stu_buf.e_sname;
    stu_var[7].datatag.str_add = stu_buf.d_birth;
    stu_var[8].datatag.str_add = stu_buf.d_cont;
    stu_var[9].datatag.str_add = stu_buf.adv_name;
    stu_var[10].datatag.str_add = stu_buf.add1;
    stu_var[11].datatag.str_add = stu_buf.add2;
    stu_var[12].datatag.str_add = stu_buf.tel;
    stu_var[13].datatag.str_add = stu_buf.p_name;
    stu_var[14].datatag.str_add = stu_buf.p_sname;
    stu_var[15].datatag.str_add = stu_buf.p_add1;
    stu_var[16].datatag.str_add = stu_buf.p_add2;
    stu_var[17].datatag.str_add = stu_buf.p_tel;
    sex_selv[0].svar_struct = ssex_var;
    ssex_var[0].datatag.str_add = ssex_buf;
    dep_selv[0].svar_struct = sdep_var;
    sdep_var[0].datatag.str_add = sdep_buf;

    /* Resume Student */
    res_var[0].datatag.str_add = res_buf.stu_id ;
    res_var[1].datatag.str_add = res_buf.locatel ;
    res_var[2].datatag.str_add = res_buf.time_int1 ;
    res_var[3].datatag.str_add = res_buf.locate2 ;
    res_var[4].datatag.str_add = res_buf.time_int2 ;
    res_var[5].datatag.str_add = res_buf.locate3 ;
    res_var[6].datatag.str_add = res_buf.time_int3 ;
    res_var[7].datatag.str_add = res_buf.locate4 ;
    res_var[8].datatag.str_add = res_buf.time_int4 ;

    /* Total Subject */
    subj_var[0].datatag.str_add = subj_buf.subj_code;
    subj_var[1].datatag.str_add = subj_buf.subj_name;
    subj_var[2].datatag.str_add = subj_buf.subj_ename;
    subj_var[3].datatag.int_add = &subj_buf.subj_cred;
    subj_var[4].datatag.int_add = &subj_buf.subj_theory;
    subj_var[5].datatag.int_add = &subj_buf.subj_opr;
    subj_var[6].datatag.str_add = subj_buf.subj_prev1;
    subj_var[7].datatag.str_add = subj_buf.subj_prev2;
    subj_var[8].datatag.str_add = subj_buf.subj_prev3;

    /* Open Subject */
    osub_var[0].datatag.str_add = osub_buf.osub_id ;
    osub_var[1].datatag.str_add = osub_buf.osub_tea ;
    osub_var[2].datatag.int_add = &osub_buf.osub_day ;
    osub_var[3].datatag.int_add = &osub_buf.osub_part1 ;
    osub_var[4].datatag.str_add = osub_buf.osub_date ;
    osub_var[5].datatag.int_add = &osub_buf.osub_part2 ;
    pa_selv[0].svar_struct = spa_var;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสำนักงานคณะกรรมการการอุดมศึกษา ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sspa1_var[0].datatag.str_add = spa1_buf;
sspa2_var[0].datatag.str_add = spa2_buf;
date_selv[0].svar_struct = sdate_var;
ssdate_var[0].datatag.str_add = sdate_buf;

/* General Subjects */
gens_var[0].datatag.int_add = &gens_buf.gens_dep ;
gens_var[1].datatag.int_add = &gens_buf.gens_year ;
gens_var[2].datatag.int_add = &gens_buf.gens_sem ;
gens_var[3].datatag.str_add = gens_buf.code ;
gsdep_var[0].datatag.str_add = gsdep_buf;
gssem_var[0].datatag.str_add = gsem_buf;

/* Select Subjects */
selct_var[0].datatag.int_add = &selct_buf.selct_dep ;
selct_var[1].datatag.int_add = &selct_buf.selct_sem ;
selct_var[2].datatag.int_add = &selct_buf.ksub ;
selct_var[3].datatag.str_add = selct_buf.slcode ;
sldep_var[0].datatag.str_add = sldep_buf ;
ssem_selv[0].svar_struct = ssem_var;
slsem_var[0].datatag.str_add = slsem_buf ;
slsub_selv[0].svar_struct = slsub_var;
ksub_var[0].datatag.str_add = slsub_buf;

/* Plan Education */
plan_var[0].datatag.int_add = &plan_buf.plan_dep ;
plan_var[1].datatag.int_add = &plan_buf.plan_sem ;
plan_var[2].datatag.int_add = &plan_buf.plan_year ;
for (i=0; i<12; i++)
    plan_var[i+3].datatag.str_add = plan_buf.sub[i].code ;
pldep_var[0].datatag.str_add = pldep_buf ;
plsem_var[0].datatag.str_add = plsem_buf ;

/* Interval Regising */
int_var[0].datatag.str_add = int_buf.st1;
int_var[1].datatag.str_add = int_buf.end1;
int_var[2].datatag.str_add = int_buf.st2;
int_var[3].datatag.str_add = int_buf.end2;

/* Change Grade For Reg Grade */
cgr_var[0].datatag.str_add = cgr_buf.cgr_sub;
cgr_var[1].datatag.str_add = cgr_buf.code;
cgr_var[2].datatag.str_add = cgr_buf.grade;

/* Mon */
mon_var[0].datatag.str_add = mon_buf.id ;
mon_var[1].datatag.int_add = &mon_buf.depp ;
mon_var[2].datatag.int_add = &mon_buf.sem ;
mon_var[3].datatag.int_add = &mon_buf.year ;
mdep_var[0].datatag.str_add = mdep_buf ;
mplsem_var[0].datatag.str_add = msem_buf ;

/* Input Grade */
inp_var[0].datatag.str_add = inp_buf.id;
inp_var[1].datatag.int_add = &inp_buf.depp ;
inp_var[2].datatag.int_add = &inp_buf.sem ;
inp_var[3].datatag.int_add = &inp_buf.year ;
for(i=0;i<12;i++)
    inp_var[4+i].datatag.str_add = inp_buf.sub[i].grd;
inpdep_var[0].datatag.str_add = inpdep_buf ;
ipsem_selv[0].svar_struct = ipsem_var;
inpsem_var[0].datatag.str_add = inpsem_buf ;

```



```

        /* Conclude Semester */
        csem_var[0].datatag.str_add = csem_buf.code ;
        csem_var[1].datatag.int_add = &csem_buf.dep ;
        csem_var[2].datatag.int_add = &csem_buf.sem ;
        csem_var[3].datatag.int_add = &csem_buf.year ;
        csemdep_var[0].datatag.str_add = csemdep_buf ;
        csemsem_var[0].datatag.str_add = csemsem_buf ;
    }
    header()
    {
        clrscr();
        printp(5,20,"ระบบ LAN กับ การประย ชาติ ๖๖๖๖",NM_HL_VDO);
        printp(7,20,"โดย",NM_HL_VDO);
        printp(9,20,"นาย โปษณ วัฒนศ ภาณุ",NM_HL_VDO);
        printp(10,20,"นาย ชัยชัย เลิศมณฑาทรรพ",NM_HL_VDO);
        printp(11,20,"นาย มนต์ชัย จรุง เรือง เกษรต",NM_HL_VDO);
        printp(12,20,"นาย สหชัย เลิศวิมลนาร",NM_HL_VDO);
        printp(14,20,"คณะ วิทยาศาสตร์",NM_HL_VDO);
        printp(15,20,"สถาบันเทคโนโลยี ภาณุ ภาณุ",NM_HL_VDO);
        printp(23,20,"กด Enter เพื่อ ใช้งาน ระบบ",NM_HL_VDO);
        while(getch() != RETURN);
    }

```



```

#include "global.inc"
#include "stuu.inc"

main ()
{
    int func_no;

    clrscr();
    set_crt(2);
    outp(0x3b8,0x28);
    outp(0x3ba,0x80);
    outp(0x3bf,0x03);
    cur_off();

    if (open_file() != 0)
    {
        bt_error();
        clrscr();
        cur_on();
        set_crt(0);
        outp(0x3ba,0x0);
        outp(0x3bf,0);
        exit(0);
    }
    header();
    assign_ptr();
    do
    {
        switch (func_no = tmenu0())
        {
            case F1:
                grade_stud();
                break;
            case F2:
                test_reg();
                break;
            case F3:
                stu_information();
                break;
        }
    } while (func_no != F9 );
    if (close_file() != 0)
        bt_error();
    clrscr();
    cur_on();
    set_crt(0);
    outp(0x3ba,0x0);
    outp(0x3bf,0);
}

tmenu0()
{
    static int func_no = F1;
    static char *t_item[] = {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อธุรกิจภายในเท่านั้น ไม่ควรเผยแพร่ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        "3. ขอบเขตของนักศึกษา",
        "9. ออกจากระบบ"
    );

    func_no = menu_func("ระบบทะเบียนนักศึกษา(อาจารย์)", F3, t_item, func_no
o);

    return(func_no);
}

grade_stud ()
{
    int func_no ;
    do
    {
        switch (func_no = tmenu1())
        {
            case F1:
                te_egrd();
                break;
            case F2:
                inputt_grade(); /*
        /*
        } while(func_no !=F9);
    }

    tmenu1()
    {
        static int func_no =F1;
        static char *t_nul[] ={
            "1. การบันทึกเกรดเฉพาะรายวิชา",
            "2. ผลการศึกษาของนักศึกษาเฉพาะราย",
            "9. ออกส่งรายการหลัก"
        };

        func_no = menu_func ("เกรดของนักศึกษา", F2, t_nul, func_no) ;
        return (func_no);
    }

    test_reg()
    {
        int func_no ;
        do
        {
            switch (func_no = tmenu2())
            {
                case F1:
                    list_sub_need(); /*
                case F2:
                    list_sub_stud();*/
                break;
            }
        } while(func_no !=F9);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

tmenu2()
{
    static int func_no = F1;
    static char *t_nu2[] = {
        "1. รายวิชาที่ต้องทำการทบทวน",
        "2. นวัตกรรมทางการศึกษา",
        "3. ออกรายการการศึกษา"
    };
    func_no = menu_func ("ตรวจสอบผลการเรียน", F2, t_nu2, func_no);
    return (func_no);
}

stu_information()
{
    int func_no ;
    do
    {
        switch (func_no = tmenu3())
        {
            case F1:
                resume_stud();
                break;
            case F2:
                re_edu();
                break;
        }
    } while(func_no != F9);
}

tmenu3()
{
    static int func_no = F1;
    static char *t_nu3[] = {
        "1. ประวัติส่วนตัว",
        "2. ประวัติการศึกษา",
        "3. ออกรายการการศึกษา"
    };
    func_no = menu_func ("การลงทะเบียนเรียน", F2, t_nu3, func_no);
    return (func_no);
}

open_file()
{
    STATUS = BTRV (B_OPEN, stu_blk, &stu_rec, &stu_len, "STU1.BTR", 1);
    if(STATUS != 0)
        return(STATUS);
    STATUS = BTRV (B_OPEN, tgr_blk, &tgr_rec, &tgr_len, "TGR.BTR", 1);
    if(STATUS != 0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN, subj_blk, &subj_rec, &subj_len, "TSUB.BTR", 1);
    if(STATUS != 0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN, osub_blk, &osub_rec, &osub_len, "OSUB.BTR", 1);
    if(STATUS != 0)
        return(STATUS) ;
}

```

เอกสารนี้เป็นเอกสารที่ STATUS = BTRV (B_OPEN, osub_blk, &osub_rec, &osub_len, "OSUB.BTR", 1); สำหรับการดำเนินการ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,res_blk,&res_rec,&res_len,"RESUME.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
if(STATUS !=0)
    return(STATUS) ;
STATUS = BTRV (B_OPEN,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
if(STATUS !=0)
    return(STATUS);
return(0);
}

close_file()
{
    STATUS = BTRV (B_CLOSE,stu_blk,&stu_rec,&stu_len,"STU1.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,tgr_blk,&tgr_rec,&tgr_len,"TGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,subj_blk,&subj_rec,&subj_len,"TSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,osub_blk,&osub_rec,&osub_len,"OSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,res_blk,&res_rec,&res_len,"RESUME.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
    if(STATUS !=0)
        return(STATUS);
    return(0);
}

assign_ptr()
{
    int i ;

    /* Student File */
    stu_var[0].datatag.str_add = stu_buf.stu_id;
    stu_var[1].datatag.int_add = &stu_buf.part;
    stu_var[2].datatag.str_add = stu_buf.th_name;
    stu_var[3].datatag.int_add = &stu_buf.sex;

```

เอกสารนี้เป็นเอกสารที่ลิขสิทธิ์เป็นของสำนักงานคณะกรรมการการอุดมศึกษา อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

stu_var[4].datatag.str_add = stu_buf.th_sname;
stu_var[5].datatag.str_add = stu_buf.ename;
stu_var[6].datatag.str_add = stu_buf.e_sname;
stu_var[7].datatag.str_add = stu_buf.d_birth;
stu_var[8].datatag.str_add = stu_buf.d_cont;
stu_var[9].datatag.str_add = stu_buf.adv_name;
stu_var[10].datatag.str_add = stu_buf.add1;
stu_var[11].datatag.str_add = stu_buf.add2;
stu_var[12].datatag.str_add = stu_buf.tel;
stu_var[13].datatag.str_add = stu_buf.p_name;
stu_var[14].datatag.str_add = stu_buf.p_sname;
stu_var[15].datatag.str_add = stu_buf.p_add1;
stu_var[16].datatag.str_add = stu_buf.p_add2;
stu_var[17].datatag.str_add = stu_buf.p_tel;

```

```

sex_selv[0].svar_struct = ssex_var;
sssex_var[0].datatag.str_add = ssex_buf;

```

```

dep_selv[0].svar_struct = sdep_var;
ssdep_var[0].datatag.str_add = sdep_buf;

```

```

/* Resume Student */
res_var[0].datatag.str_add = res_buf.stu_id ;
res_var[1].datatag.str_add = res_buf.locatel ;
res_var[2].datatag.str_add = res_buf.time_int1 ;
res_var[3].datatag.str_add = res_buf.locate2 ;
res_var[4].datatag.str_add = res_buf.time_int2 ;
res_var[5].datatag.str_add = res_buf.locate3 ;
res_var[6].datatag.str_add = res_buf.time_int3 ;
res_var[7].datatag.str_add = res_buf.locate4 ;
res_var[8].datatag.str_add = res_buf.time_int4 ;

```

```

/* Total Subject */
subj_var[0].datatag.str_add = subj_buf.subj_code;
subj_var[1].datatag.str_add = subj_buf.subj_name;
subj_var[2].datatag.str_add = subj_buf.subj_ename;
subj_var[3].datatag.int_add = &subj_buf.subj_cred;
subj_var[4].datatag.int_add = &subj_buf.subj_theory;
subj_var[5].datatag.int_add = &subj_buf.subj_opr;
subj_var[6].datatag.str_add = subj_buf.subj_prev1;
subj_var[7].datatag.str_add = subj_buf.subj_prev2;
subj_var[8].datatag.str_add = subj_buf.subj_prev3;

```

```

/* Open Subject */
/*      osub_var[0].datatag.str_add = osub_buf.osub_id;
      osub_var[1].datatag.str_add = osub_buf.osub_tea;
      osub_var[2].datatag.int_add = &osub_buf.osub_day;
      osub_var[3].datatag.int_add = &osub_buf.osub_part1;
      osub_var[4].datatag.str_add = osub_buf.osub_date;
      osub_var[5].datatag.int_add = &osub_buf.osub_part; */

```

```

/*      pa_selv[0].svar_struct = spa_var;
      sspal_var[0].datatag.str_add = spal_buf;
      pa_selv[0].svar_struct = spa_var;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยราชภัฏวไลยอลงกรณ์
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sspa_var[0].datatag.str_add = spa_buf;
date_selv[0].svar_struct = sdate_var;
ssdate_var[0].datatag.str_add = sdate_buf;*/

/* Input Grade */
inp_var[0].datatag.str_add = inp_buf.id;
for(i=0;i<12;i++)
    inp_var[i+1].datatag.str_add = inp_buf.sub[i].grd;

/* Teacher Input Grade */
tgr_var[0].datatag.str_add = tgr_buf.tgr_sub ;
tgr_var[1].datatag.str_add = tgr_buf.code ;
tgr_var[2].datatag.str_add = tgr_buf.grade1 ;
tgr_var[3].datatag.str_add = tgr_buf.grade2 ;

/* Resume Student */
res_var[0].datatag.str_add = res_buf.stu_id;
res_var[1].datatag.str_add = res_buf.locate1;
res_var[2].datatag.str_add = res_buf.time_int1;
res_var[3].datatag.str_add = res_buf.locate2;
res_var[4].datatag.str_add = res_buf.time_int2;

/* Mon */
mon_var[0].datatag.str_add = mon_buf.id ;
mon_var[1].datatag.int_add = &mon_buf.depp ;
mon_var[2].datatag.int_add = &mon_buf.sem ;
mon_var[3].datatag.int_add = &mon_buf.year ;
mdep_var[0].datatag.str_add = mdep_buf ;
mplsem_var[0].datatag.str_add = msem_buf ;

/* Input Grade */
inp_var[0].datatag.str_add = inp_buf.id;
inp_var[1].datatag.int_add = &inp_buf.depp ;
inp_var[2].datatag.int_add = &inp_buf.sem ;
inp_var[3].datatag.int_add = &inp_buf.year ;
for(i=0;i<12;i++)
    inp_var[4+i].datatag.str_add = inp_buf.sub[i].grd;
inpdep_var[0].datatag.str_add = inpdep_buf ;
ipsem_selv[0].svar_struct = ipsem_var;
inpsem_var[0].datatag.str_add = inpsem_buf ;
}

header()
{
    clrscr();
    printp(5,20,"ระบบ LAN การลงทะเบียน รายวิชา",NM_HL_VDO);
    printp(7,20,"โดย",NM_HL_VDO);
    printp(9,20,"นาย โปษณัย พจนศกตสกร",NM_HL_VDO);
    printp(10,20,"นาย ชัชชัย เลิศมณฑาทรรพ",NM_HL_VDO);
    printp(11,20,"นาย มนต์ชัย จรุงเรืองเกียรติ",NM_HL_VDO);
    printp(12,20,"นาย สุธชัย เลิศวัฒนาภ",NM_HL_VDO);
    printp(14,20,"คณะ วิทยาศาสตร์",NM_HL_VDO);
    printp(15,20,"สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง",NM_HL_VDO);
}

```



```
printp(23,20,"กด Enter เพื่อเปลี่ยนระบบ",NM_HL_VDO);  
while(getch() != RETURN);  
}
```



```
#include "global.inc"
#include "stuu.inc"
main ()
{
    int func_no;
    clrscr();
    set_crt(2);
    outp(0x3b8,0x28);
    outp(0x3ba,0x80);
    outp(0x3bf,0x03);
    cur_off();

    if(open_file() != 0)
    {
        bt_error();
        clrscr();
        cur_on();
        set_crt(0);
        outp(0x3ba,0x0);
        outp(0x3bf,0);
        exit(0);
    }

    header() ;
    assign_ptr();

    do
    {
        switch (func_no =menu0())
        {
            case F1:
                re_sume();
                break;
            case F2:
                result_edu();
                break;
            case F3:
                conclude_reg();
                break;
            case F4:
                des_sub();
                break;
        }
    } while (func_no !=F9 );
    if (close_file() != 0)
    {
        bt_error();
        clrscr();
        cur_on();
        set_crt(0) ;
        outp(0x3ba,0x0);
        outp(0x3bf,0);
    }
}
```

```

static int func_no = F1;
static char *m_item[] = {
    "1.ประมวลผลจำนวนตัว",
    "2.ตรวจสอบผลการศีกษา",
    "3.ลงทะเบียนและแฟ้ม,ก่อนเข้าเรียน",
    "4.ตรวจสอบรายละเอียดวิชาที่ไปทดสอบ",
    "9.ออกจากระบบ"
};

func_no = menu_func("ระบบทะเบียนนักศึกษา(นักศึกษา)", F4, m_item, func_no);

return(func_no);
}

result_edu()
{
    int func_no ;
    do
    {
        switch (func_no = menu2())
        {
            case F1:
                test_sub();
                break;
            case F2:
                /*ggister();*/
                break;
            case F3:
                /*te_sub();*/
                break;
        }
    } while(func_no != F9);
}

menu2()
{
    static int func_no = F1;
    static char *m_nu2[] = {
        "1.ตรวจสอบผลเป็นรายวิชา",
        "2.ตรวจสอบผลเป็นปการศึกษา",
        "3.สรุผลการศึกษาจนจบปัจจุบัน",
        "9.ออกสง,รายการหลก"
    };

    func_no = menu_func ("ตรวจสอบผลการศีกษา", F3, m_nu2, func_no);
    return (func_no);
}

des_sub()
{
    int func_no ;
    do
    {
        switch (func_no = menu4())
        {
            case F1:
                open_major();

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    case F2:
        gen_subject();
        break;
    case F3:
        selct_subject();
        break;
    case F4:
        open_subject() ;
        break;
    }
} while(func_no !=F9);
}

menu4()
{
    static int func_no =F1;
    static char *m_nu4[] = {
        "1.วิชาทั้งหมดด้านหลักสูตร",
        "2.วิชาบังคับเฉพาะภาคการศึกษา",
        "3.วิชาเลือกเฉพาะภาคการศึกษา",
        "4.วิชาที่เปิดสอนในปัจจุบัน",
        "9.ออกสจ.รายการหลักสูตร"
    };
    func_no = menu_func ("รายละเอียดของรายวิชา",F4,m_nu4,func_no);
    return (func_no);
}

open_file()
{
    STATUS = BTRV (B_OPEN,stu_blk,&stu_rec,&stu_len,"STU1.BTR",1);
    if(STATUS !=0)
        return(STATUS);
    STATUS = BTRV (B_OPEN,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
    if(STATUS !=0)
        return(STATUS);
    STATUS = BTRV (B_OPEN,osub_blk,&osub_rec,&osub_len,"OSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,subj_blk,&subj_rec,&subj_len,"TSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,res_blk,&res_rec,&res_len,"RESUME.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_OPEN,gens_blk,&gens_rec,&gens_len,"GENSUB.BTR",1);
    if(STATUS !=0)
        return (STATUS) ;
}

```

เอกสารนี้เป็นเอกสารที่...การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        STATUS = BTRV (B_OPEN,selct_blk,&selct_rec,&selct_len,"SELECT.BTR",
1);
        if(STATUS !=0)
            return(STATUS) ;
        STATUS = BTRV (B_OPEN,tgr_blk,&tgr_rec,&tgr_len,"TGR.BTR",1);
        if(STATUS !=0)
            return(STATUS) ;

        return(0);
    }

close_file()
{
    STATUS = BTRV (B_CLOSE,stu_blk,&stu_rec,&stu_len,"STU1.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,inp_blk,&inp_rec,&inp_len,"GRADE.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,osub_blk,&osub_rec,&osub_len,"OSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,subj_blk,&subj_rec,&subj_len,"TSUB.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,mon_blk,&mon_rec,&mon_len,"MON.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,plan_blk,&plan_rec,&plan_len,"PLAN.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,res_blk,&res_rec,&res_len,"RESUME.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,gens_blk,&gens_rec,&gens_len,"GENSUB.BTR",1)
;
    if(STATUS !=0)
        return (STATUS) ;
    STATUS = BTRV (B_CLOSE,selct_blk,&selct_rec,&selct_len,"SELECT.BTR"
,1);
    if(STATUS !=0)
        return(STATUS) ;
    STATUS = BTRV (B_CLOSE,tgr_blk,&tgr_rec,&tgr_len,"TGR.BTR",1);
    if(STATUS !=0)
        return(STATUS) ;

    return(0);
}

assign_ptr()
{
    int i ;
    /* Student File */

```

```

stu_var[0].datatag.str_add = stu_buf.stu_id;
stu_var[1].datatag.int_add = &stu_buf.part;
stu_var[2].datatag.str_add = stu_buf.th_name;
stu_var[3].datatag.int_add = &stu_buf.sex;
stu_var[4].datatag.str_add = stu_buf.th_sname;
stu_var[5].datatag.str_add = stu_buf.ename;
stu_var[6].datatag.str_add = stu_buf.e_sname;
stu_var[7].datatag.str_add = stu_buf.d_birth;
stu_var[8].datatag.str_add = stu_buf.d_cont;
stu_var[9].datatag.str_add = stu_buf.adv_name;
stu_var[10].datatag.str_add = stu_buf.add1;
stu_var[11].datatag.str_add = stu_buf.add2;
stu_var[12].datatag.str_add = stu_buf.tel;
stu_var[13].datatag.str_add = stu_buf.p_name;
stu_var[14].datatag.str_add = stu_buf.p_sname;
stu_var[15].datatag.str_add = stu_buf.p_add1;
stu_var[16].datatag.str_add = stu_buf.p_add2;
stu_var[17].datatag.str_add = stu_buf.p_tel;

sex_selv[0].svar_struct = ssex_var;
sssex_var[0].datatag.str_add = ssex_buf;

dep_selv[0].svar_struct = sdep_var;
ssdep_var[0].datatag.str_add = sdep_buf;

/* Total Subject */
subj_var[0].datatag.str_add = subj_buf.subj_code;
subj_var[1].datatag.str_add = subj_buf.subj_name;
subj_var[2].datatag.str_add = subj_buf.subj_ename;
subj_var[3].datatag.int_add = &subj_buf.subj_cred;
subj_var[4].datatag.int_add = &subj_buf.subj_theory;
subj_var[5].datatag.int_add = &subj_buf.subj_opr;
subj_var[6].datatag.str_add = subj_buf.subj_prev1;
subj_var[7].datatag.str_add = subj_buf.subj_prev2;
subj_var[8].datatag.str_add = subj_buf.subj_prev3;

/* Test result education */
subj_var[0].datatag.str_add = subj_buf.subj_code;
subj_var[1].datatag.int_add = &subj_buf.subj_cred;

/*subj_var[2].datatag.str_add = subj_buf.subj_date;
subje_var[3].datatag.str_add = subj_buf.subj_grade;*/

/* Total Subject Back_up*/
/* sbk_var[0].datatag.str_add = subj_babuf.subj_code;
sbk_var[1].datatag.str_add = subj_babuf.act;
sbk_var[2].datatag.str_add = subj_babuf.date;
sbk_var[3].datatag.str_add = subj_babuf.time; */

/* Open Subject */
osub_var[0].datatag.str_add = osub_buf.osub_id ;
osub_var[1].datatag.str_add = osub_buf.osub_tea ;
osub_var[2].datatag.int_add = &osub_buf.osub_day ;

```



```

osub_var[3].datatag.int_add = &osub_buf.osub_part1 ;
osub_var[4].datatag.str_add = osub_buf.osub_date ;
osub_var[5].datatag.int_add = &osub_buf.osub_part2 ;

```

```

pa_selv[0].svar_struct = spa_var;
sspa1_var[0].datatag.str_add = spa1_buf;
sspa2_var[0].datatag.str_add = spa2_buf;
date_selv[0].svar_struct = sdate_var;
ssdate_var[0].datatag.str_add = sdate_buf;

```

```

/* Input Grade */

```

```

inp_var[0].datatag.str_add = inp_buf.id;
for(i=0;i<12;i++)
inp_var[i+1].datatag.str_add = inp_buf.sub[i].grd;

```

```

/* Mon */

```

```

mon_var[0].datatag.str_add = mon_buf.id ;
mon_var[1].datatag.int_add = &mon_buf.depp ;
mon_var[2].datatag.int_add = &mon_buf.sem ;
mon_var[3].datatag.int_add = &mon_buf.year ;
mdep_var[0].datatag.str_add = mdep_buf ;
mplsem_var[0].datatag.str_add = msem_buf ;

```

```

/* Plan Education */

```

```

plan_var[0].datatag.int_add = &plan_buf.plan_dep ;
plan_var[1].datatag.int_add = &plan_buf.plan_sem ;
plan_var[2].datatag.int_add = &plan_buf.plan_year ;
for (i=0; i<12; i++)
plan_var[i+3].datatag.str_add = plan_buf.sub[i].code ;
pldep_var[0].datatag.str_add = pldep_buf ;
plsem_var[0].datatag.str_add = plsem_buf ;

```

```

/* Resume Student */

```

```

res_var[0].datatag.str_add = res_buf.stu_id;
res_var[1].datatag.str_add = res_buf.locate1;
res_var[2].datatag.str_add = res_buf.time_int1;
res_var[3].datatag.str_add = res_buf.locate2;
res_var[4].datatag.str_add = res_buf.time_int2;

```

```

/* Interval Regising */

```

```

int_var[0].datatag.str_add = int_buf.st1;
int_var[1].datatag.str_add = int_buf.end1;
int_var[2].datatag.str_add = int_buf.st2;
int_var[3].datatag.str_add = int_buf.end2;

```

```

/* General Subjects */

```

```

gens_var[0].datatag.int_add = &gens_buf.gens_dep;
gens_var[1].datatag.int_add = &gens_buf.gens_year;
gens_var[2].datatag.int_add = &gens_buf.gens_sem;
gens_var[3].datatag.str_add = gens_buf.code;

```

```

gsdep_var[0].datatag.str_add = gsdep_buf;
ssem_selv[0].svar_struct = ssem_var;
gssem_var[0].datatag.str_add = gsem_buf;

/* Select Subjects */
selct_var[0].datatag.int_add = &selct_buf.selct_dep;
selct_var[1].datatag.int_add = &selct_buf.selct_ssem;
selct_var[2].datatag.int_add = &selct_buf.ksub;
selct_var[3].datatag.str_add = selct_buf.slcode;

sldep_var[0].datatag.str_add = sldep_buf;
slsem_var[0].datatag.str_add = slsem_buf;
slsub_selv[0].svar_struct = slsub_var;
ksub_var[0].datatag.str_add = slsub_buf;

/* Teacher Input Grade */
tgr_var[0].datatag.str_add = tgr_buf.tgr_sub;
tgr_var[1].datatag.str_add = tgr_buf.code;
tgr_var[2].datatag.str_add = tgr_buf.gradel;
tgr_var[3].datatag.str_add = tgr_buf.grade2;

}
header()
{
clrscr();
printf(5,23,"ระบบ LAN การประชุม ทดสอบงาน",NM_HL_VDO);
printf(7,34,"โดย",NM_HL_VDO);
printf(9,25,"นาย ไพพงษ์ พงษ์ศิริ",NM_HL_VDO);
printf(10,25,"นาย ชัยชัย เลิศมณฑาทิพย์",NM_HL_VDO);
printf(11,25,"นาง มณฑะ จรุงเรืองเกียรติ",NM_HL_VDO);
printf(12,25,"นาย สหชัย เลิศวิมลารักษ์",NM_HL_VDO);
printf(15,30,"คณะ วิทยาศาสตร์",NM_HL_VDO);
printf(16,20,"สถาบันเทคโนโลยี จุฬาลงกรณ์มหาวิทยาลัย",NM_HL_VDO);
printf(23,25,"กด Enter เพื่อเข้าสู่ระบบ",NM_HL_VDO);
while(getch() != RETURN);
}

```

```

#include "extern.inc"
#include "stue.inc"

int    total ;

conclude_reg ()
{
    int    func_no ;

    mon_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
    if (STATUS==0)
    {
        mon_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
        mon_clr0 () ;
        mon_print () ;
    }
    do {
        switch (func_no = mon_menu())
        {
            case F1:
                mon_ins () ;
                break ;
            case F2:
                mon_del () ;
                break ;
            case F3:
                mon_oth () ;
                break ;
        }
    } while (func_no != F9) ;
}

mon_plan2mon ()
{
    int    i ;

    mon_buf.depp= plan_rec.plan_dep;
    mon_buf.sem = plan_rec.plan_sem ;
    mon_buf.year = plan_rec.plan_year ;
    for (i=0; i<12; i++)
        strcpy (mon_buf.sub[i].code,plan_rec.sub[i].code) ;
}

mon_rec2buf ()
{
    int    i ;

    strcpy (mon_buf.id,mon_rec.id) ;
    mon_buf.depp = mon_rec.depp ;
    mon_buf.sem = mon_rec.sem ;
    mon_buf.year = mon_rec.year ;
    for (i=0; i<12; i++)

```

```

    {
        strcpy (mon_buf.sub[i].code,mon_rec.sub[i].code) ;
    }
}

mon_buf2rec ()
{
    int    i ;

    strcpy (mon_rec.id,mon_buf.id) ;
    mon_rec.depp = mon_buf.depp ;
    mon_rec.sem = mon_buf.sem ;
    mon_rec.year = mon_buf.year ;
    for (i=0; i<12; i++)
    {
        mon_buf.sub[i].code[6] = '\0' ;
        strcpy (mon_rec.sub[i].code,mon_buf.sub[i].code) ;
    }
}

mon_scr ()
{
    clrscr () ;
    frame ( 2, 1,20,79,"ระบบป้อนข้อมูลการลงทะเบียน") ;
    lineH ( 1,79, 7) ;
    lineV ( 7,20,10) ;
    lineV ( 7,20,69) ;
    printp ( 4,13,"ชื่อ : ",NM_VDO) ;
    printp ( 5, 9,"ภาควิชา : ",NM_VDO) ;
    printp ( 7, 3,"รหัสวิชา",RV_VDO) ;
    printp ( 7,32,"ชื่อวิชา",RV_VDO) ;
    printp ( 7,71,"หน่วยกิต",RV_VDO) ;
    mon_clr0();
    printp (21,66,"รวม : ",NM_VDO) ;
    printp (21,72,"0",RV_VDO) ;
}

mon_menu ()
{
    static struct lbl_template    fn_mon[] = {
        {23, 4,"1: เพิ่มข้อมูล",RV_VDO
    },
        {23,16,"2: ลบ",RV_VDO},
        {23,24,"3: นุ้ศกษากอน",RV_
VDO},
        {23,71,"9: Exit",RV_VDO}
    } ;

    int    i ;

    i = get_func (F3,fn_mon) ;
    return (i) ;
}

```

```

{
    int    i ;
    mon_name();
    mon_dep();
    mon_sem();
    total = 0 ;
    for (i=0; i<12; i++)
    {
        if (mon_buf.sub[i].code[0] != ' ')
            mon_getsub (i) ;
    }
    mon_total () ;
}

mon_name ()
{
    int    i ;

    strcpy (stu_key0.id,mon_buf.id) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบรหัสนักศึกษาในแฟ้มข้อมูล") ;
        return (FALSE) ;
    }
    printp ( 4,18,"",NM_VDO) ;
    ;
    printp ( 4,18,stu_rec.th_name,NM_HL_VDO) ;
    i = 0 ;
    while (stu_rec.th_name[i] != ' ')
        i++ ;
    printp ( 4,20+i,stu_rec.th_sname,NM_HL_VDO) ;
    return (TRUE) ;
}

mon_total ()
{
    char    tbuf[3] ;

    printp (21,72,"",NM_VDO) ;
    itoa_1 (total,tbuf,2) ;
    printp (21,72,tbuf,NM_HL_VDO) ;
}

mon_clr0 ()
{
    int    i ;

    for (i=0; i<12; i++)
    {
        printp (8+i,3,"",NM_VDO) ;
        printp (8+i,12,"",NM_VD
0) ;
        printp (8+i,71,"",NM_VDO) ;
    }
}

```

```

    printp (21,72," ",NM_VDO) ;
}

mon_getsub (int i)
{
    char    gbuf[2] ;
    char    gbuf1[61] ;
    int     n ;

    mon_buf.sub[i].code[6] = '\0' ;
    strcpy (osub_key0.osub_id,mon_buf.sub[i].code) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบการเปิดสอนกระบวนวิชา") ;
        return (FALSE) ;
    }
    strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบกระบวนวิชาในแฟ้มข้อมูล") ;
        return (FALSE) ;
    }
    strcpy (gbuf1,subj_rec.subj_name) ;
    n = 0 ;
    while (gbuf1[n] != ' ' || gbuf1[n+1] != ' ')
        n++ ;
    gbuf1[n] = '\0' ;
    printp (8+i,3,subj_rec.subj_code,NM_VDO) ;
    printp (8+i,14,"",NM_VDO) ;

    printp (8+i,14,gbuf1,NM_HL_VDO) ;
    itoa_1 (subj_rec.subj_cred,gbuf,1) ;
    printp (8+i,73,gbuf,NM_HL_VDO) ;
    total += subj_rec.subj_cred ;
    return (TRUE) ;
}

depp(c,b)
int c;
char b[];
{
    static char *dep[] = {"คณิตศาสตร์และวิทยาการคอมพิวเตอร์",
        "เคมีอุตสาหกรรม",
        "เทคโนโลยีชีวภาพ",
        "ฟิสิกส์ประยุกต์",
        "สถิติประยุกต์"} ;

    strcpy(b,dep[c]);
}

mon_dep ()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

depp(mon_buf.depp,part2);
printp(mon_var[1].row,mon_var[1].col,part2,NM_HL_VDO);
printp (5,18,"",NM_VDO);
printp (5,18,part2,NM_HL_VDO);
return (TRUE) ;
}

semm(cd,bb)
int cd;
char bb[];
{
static char *sem[] = {"1st",
                      "2nd",
                      "Summer" } ;
strcpy(bb,sem[cd]);
}

mon_sem ()
{
char sbuf[7] ;

semm(mon_buf.sem,sbuf);
printp(mon_var[2].row,mon_var[2].col,sbuf,NM_HL_VDO);
printp(6,18,"",NM_VDO);
printp( 6,18,sbuf,NM_HL_VDO) ;
return(TRUE);
}

mon_year ()
{
char ybuf[2] ;

printp ( 6,38,"",NM_VDO) ;
itoa_1 (mon_buf.year,ybuf,1) ;
printp ( 6,38,ybuf,NM_HL_VDO) ;
}

mon_ins ()
{
static struct var_template cins_var[] = {
                                {19,40,STRING,"99-999","99-999",}
                                } ;

static struct lbl_template cins_lbl[] = {
                                {19,31,"รหัสเข้า :",NM_VDO}
                                } ;

int status, dflag = -1 ;
char dbuf[7], gbuf[2] ;
int subptr = 0,i ;

if (mon_buf.id[0] == ' ')
{

```



```

static struct lbl_template cdel_lbl[] = {
    {19,31,"รหัสวิชา :",NM_VDO}
};

int status, i, dflag;
char dbuf[7];

if (mon_buf.id[0] == ' ')
{
    error ("ต้องใส่รหัสวิชาจากบน");
    return;
}

cdel_var[0].datatag.str_add = dbuf;
make_win (16,29,21,50," ");
printp (18,31,"ป้อนรหัสวิชาที่ต้องการลบ",NM_VDO);
scr_mgr (NEW,DISP_FLAG,NEW,cdel_var,cdel_lbl,1,NULL,NULL);
T_E_flag = Eng;
cur_fld = 0;
do {
    status = scr_mgr (OLD,ED_FLAG,OLD,cdel_var,cdel_lbl,1,NULL,
NULL);
} while (status != F10 && status != ESC && status != RETURN);
recall ();
if (status == ESC)
    return;
dflag = -1;
dbuf[6] = '\0';
for (i=0; i<12; i++)
{
    if (strcmp (mon_buf.sub[i].code,dbuf)==0)
        dflag = i;
}
if (dflag == -1)
{
    error ("ยังไม่ได้ลงทะเบียนวิชา");
    return;
}
strcpy (subj_key0.subj_code,mon_buf.sub[dflag].code);
STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
total -= subj_rec.subj_cred;
strcpy (mon_buf.sub[dflag].code,"");
mon_sortsub ();
mon_buf2rec ();
STATUS = BTRV (B_UPDATE,mon_blk,&mon_rec,&mon_len,&mon_key0,0);
if (STATUS != 0)
    b_error ("ไม่สามารถลบข้อมูล");
scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL);
mon_clr0 ();
mon_print ();
}

mon_oth ()
{
    int status, st, sure, i, STAT1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
clr_win (5,18,5,50) ;
clr_win (6,18,6,22) ;
clr_win (8,11,19,68) ; clr_win (8,70,19,78) ;
printp (4,18,"",NM_VDO)
;
mon_clr0();
printp (21,58," ",NM_VDO) ;
T_E_flag = Eng ;
cur_fld = 0 ;
do {
    if (cur_fld==1)
    {
        T_E_flag = Eng;
        status = fld_sel (&dep_var[0],&dep_selv[0],dep_sel
v[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
        .mon_buf.depp = dep_selv[0].cur_sv ;
        cur_fld = 2 ;
    }
    if (cur_fld==2)
    {
        T_E_flag = Eng;
        status = fld_sel (&plsem_var[0],&ssem_selv[0],ssem
_selv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
        mon_buf.sem = ssem_selv[0].cur_sv ;
        if (status==CUR_UP)
        cur_fld = 1 ;
        else
        cur_fld = 3 ;
    }
    status = scr_mgr (OLD,ED_FLAG,OLD,mon_var,mon_lbl,4,NULL,NU
LL) ;
    mon_fld (&status);
    if (status == F10 || old_fld == 3)
    {
        plan_key0.plan_dep = mon_buf.depp;
        plan_key0.plan_sem = mon_buf.sem;
        plan_key0.plan_year = mon_buf.year ;
        STATUS = BTRV (B_GETEQ,plan_blk,&plan_rec,&plan_len
,&plan_key0,0) ;
        if (STATUS==0)
        {
            mon_buf.depp= plan_rec.plan_dep;
            mon_buf.sem = plan_rec.plan_sem ;
            mon_buf.year = plan_rec.plan_year ;
            for (i=0; i<12; i++)
                strcpy (mon_buf.sub[i].code,plan_re
c.sub[i].code) ;
            mon_print () ;
        }
        else {
            b_error("ย.ไม่ ออกแบบโครงสร้างหลก๓๓๓๓") ;
            cur_fld = 0 ;
            status = RETURN ;
        }
    }
}

```

```

    }
}
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    STATUS = BTRV (B_GETPR,mon_blk,&mon_rec,&mon_len,&mon_key0,
0) ;

    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,mon_blk,&mon_rec,&mon_len,&mon_key0,
0) ;

    if (STATUS == 0)
    {
        mon_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,N
ULL) ;

        mon_print () ;
    }
    return ;
}
strcpy (mon_key0.id,mon_buf.id) ;
mon_key0.depp = mon_buf.depp ;
mon_key0.sem = mon_buf.sem ;
mon_key0.year = mon_buf.year ;
STAT1 = BTRV (B_GETEQ,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
if (STAT1 == 0)
{
    b_error ("มีข้อมูลน้อยเกินไปหรือไม่") ;
    mon_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
    mon_print () ;
    return ;
}
sure = make_sure ("โปรดยืนยันการพิมพ์เพิ่มเติมหรือไม่") ;
if (sure==0)
    return ;
mon_buf2rec () ;
STATUS = BTRV (B_INSERT,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถเก็บข้อมูลได้") ;
    STATUS = BTRV (B_GETPR,mon_blk,&mon_rec,&mon_len,&mon_key0,
0) ;

    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;

    scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
    printp (4,18,"",NM_V
DO) ;

    if (STATUS == 0)
    {
        mon_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,N
ULL) ;

        mon_print () ;
    }
}

```

```

        return ;
    }
    scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
    mon_clr0 () ;
    mon_print () ;
}

mon_sortsub ()
{
    char    scode[12][6] ;
    int     tmptmp ;
    int     point1, point2 ;
    int     tmptag[12] ;

    for (point1=0; point1<12; point1++)
    {
        strcpy (scode[point1],mon_buf.sub[point1].code) ;
        tmptag[point1] = point1 ;
    }
    for (point1=0; point1<11; point1++)
    {
        for (point2=point1+1; point2<12; point2++)
        {
            if (strcmp(scode[tmptag[point1]],scode[tmptag[point
2]]) > 0)
            {
                tmptmp = tmptag[point1] ;
                tmptag[point1] = tmptag[point2] ;
                tmptag[point2] = tmptmp ;
            }
        }
        point1 = 0 ;
        while (scode[tmptag[point1]][0] == ' ')
            point1++ ;
        while (point1 < 12)
        {
            tmptmp = tmptag[11] ;
            for (point2=10; point2 >= 0; point2--)
                tmptag[point2+1] = tmptag[point2] ;
            tmptag[0] = tmptmp ;
            point1++ ;
        }
        for (point1=0; point1<12; point1++)
            strcpy (mon_buf.sub[point1].code,scode[tmptag[point1]]) ;
    }
}

mon_fld (stat)
int *stat;
{
    if (*stat == ESC)
        return ;
    if (old_fld == 0 || *stat == F10)
        {
            if ((mon_name()) == FALSE)
                return ;
            if ((mon_name()) == FALSE)
                return ;
        }
}

```



```
{
    *stat = RETURN;
    cur_fld = 0 ;
    return;
}
}
if (old_fld ==1 && *stat == F10)
{
    if (mon_buf.depp > 4 )
    {
        error("โปรดระวังค่า") ;
        cur_fld = old_fld ;
        *stat = RETURN ;
        return;
    }
}
if (old_fld ==2 && *stat == F10)
{
    if (mon_buf.sem > 2)
    {
        error ("โปรดระวังเพศ") ;
        cur_fld = old_fld ;
        *stat = RETURN ;
        return;
    }
}
}
```



```

#include "extern.inc"
#include "stue.inc"

open_subject ()
{
    int      func_no ;

    osub_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,osub_var,osub_lbl,6,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS==0)
    {
        osub_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
        osub_print () ;
    }
    do
    {
        switch (func_no = osub_menu())
        {
            case F1:
                osub_prev () ;
                break ;
            case F2:
                osub_next () ;
                break ;
            case F3:
                osub_srch () ;
                break ;
        }
    } while (func_no!=F9) ;
}

osub_scr ()
{
    clrscr () ;
    frame (9,5,18,80,"ระบบบงกชของกรมการนิเทศและป้องกันควบคุมโรคติดต่อ  

);
    printf(12,14,"ชื่ออาสา :",NM_VDO) ;
    printf(13,7,"ชื่ออาสา(อังกฤษ) :",NM_VDO);
    printf(16,29,"[dd/mm]",NM_VDO) ;
}

osub_rec2buf ()
{
    strcpy (osub_buf.osub_id,osub_rec.osub_id) ;
    strcpy (osub_buf.osub_tea,osub_rec.osub_tea) ;
    osub_buf.osub_day = osub_rec.osub_day ;
    osub_buf.osub_part1 = osub_rec.osub_part1 ;
    strcpy (osub_buf.osub_date,osub_rec.osub_date) ;
    osub_buf.osub_part2 = osub_rec.osub_part2 ;
}

osub_buf2rec ()

```

```

        strcpy (osub_rec.osub_id,osub_buf.osub_id) ;
        strcpy (osub_rec.osub_tea,osub_buf.osub_tea) ;
        osub_rec.osub_day = osub_buf.osub_day ;
        osub_rec.osub_part1 = osub_buf.osub_part1 ;
        strcpy (osub_rec.osub_date,osub_buf.osub_date) ;
        osub_rec.osub_part2 = osub_buf.osub_part2 ;
    }

osub_print ()
{
    osub_name () ;
    select_date () ;
    osub_part (osub_buf.osub_part1,15,58) ;
    osub_date (osub_buf.osub_date,16,38) ;
    osub_part (osub_buf.osub_part2,16,58) ;
}

osub_name ()
{
    char    nsub[61] ;
    char    ensub[61];

    strcpy (subj_key0.subj_code,osub_buf.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS!=0)
    {
        bt_error () ;
        return (FALSE) ;
    }
    strcpy (nsub,subj_rec.subj_name) ;
    strcpy (ensub,subj_rec.subj_ename) ;
    nsub[49] = '\0' ;
    ensub[55] = '\0' ;
    printp (12,22,"                                ",NM_VD
0) ;
    printp (12,22,nsub,NM_HL_VDO) ;
    printp (13,22,"                                "
,NM_VDO) ;
    printp (13,22,ensub,NM_HL_VDO) ;
    return (TRUE) ;
}

select_date ()
{
    static char *day[] = {"จันทร์",
                           "อังคาร",
                           "พุธ",
                           "พฤหัสบดี",
                           "ศุกร์",
                           "เสาร์",
                           "อาทิตย์"};

    char    dbuf[9] ;
    strcpy (dbuf,day[osub_buf.osub_day]) ;

```



```

        return (FALSE) ;
    else
        return (TRUE) ;
}

osub_part (pa,a,b)
int      pa,a,b ;
{
    static char      *part[] = { "เซต", "บว", "ลบ" } ;

    printp (a,b,"      ",NM_VDO) ;
    printp (a,b,part[pa],NM_HL_VDO) ;
}

osub_menu ()
{
    static struct lbl_template fn_osub[] = { {23, 4,"1:ก.อน",RV_VDO},
                                              {23,20,"2:ผล.1",RV_VDO},
                                              {23,35,"3:ค.พท",RV_VDO},
                                              {23,53,"9:Exit",RV_VDO}
                                              } ;

    int      i ;

    i = get_func (F3,fn_osub) ;
    return (i) ;
}

osub_prev ()
{
    STATUS = BTRV (B_GETPR,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ก.ง.ขอมีผลแรกแล้วครับ") ;
        return ;
    }
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
}

osub_next ()
{
    STATUS = BTRV (B_GETNX,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ก.ง.ขอมีผลสุดท้ายแล้วครับ") ;
        return ;
    }
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
}

osub_srch ()
{
    int      status, new, row, col ;

```

```

char    buff[10] ;

T_E_flag = Eng ;
make_win (18, 6,22,36,"Search Window") ;
printp (20,12,"รหัสนี้ : ",NM_HL_VDO) ;
strcpy (buff,osub_buf.osub_id) ;
row = osub_var[0].row ;
col = osub_var[0].col ;
osub_var[0].row = 20 ;
osub_var[0].col = 21 ;
new = NEW ;
do
{
    status = fld_edit (new,&osub_var[0],NAUTO_ADV,NULL,NULL) ;
    new = OLD ;
} while (status != RETURN && status != F10 && status != ESC) ;
osub_var[0].row = row ;
osub_var[0].col = col ;
recall () ;
if (status == ESC)
{
    strcpy (osub_buf.osub_id,buff) ;
    return ;
}
strcpy (osub_key0.osub_id,osub_buf.osub_id) ;
STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่พบรหัสนี้ในแฟ้มข้อมูล") ;
    return ;
}
osub_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
osub_print () ;
}

```

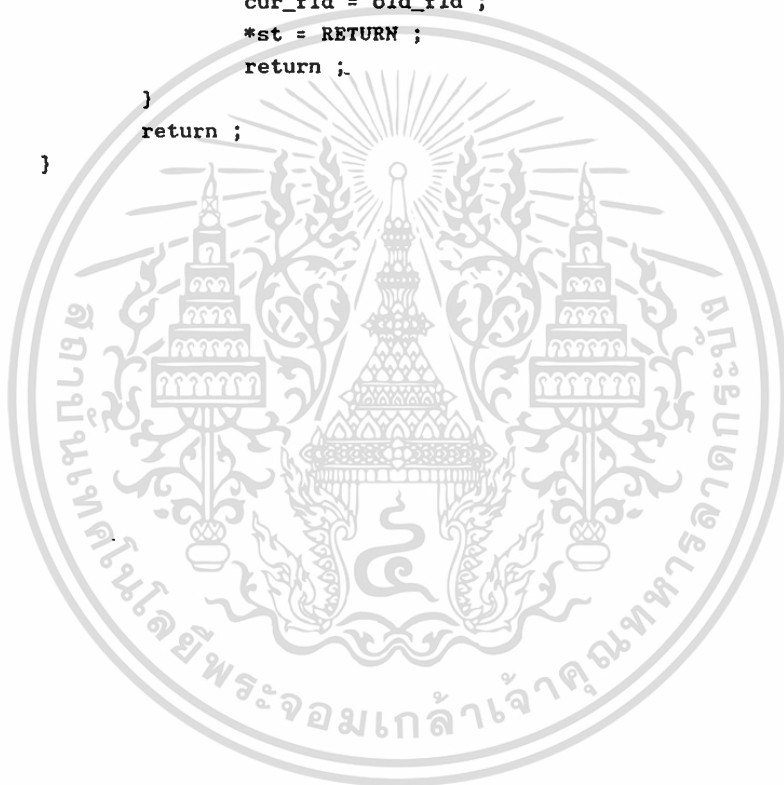
```

osub_fld (st)
int    *st ;
{
    if (*st==ESC)
        return ;
    if (old_fld==0 && cur_fld!=0)
    {
        if (osub_name ()==FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
        return ;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
{
    cur_fld = old_fld ;
    *st = RETURN ;
    return ;
}
return ;
}
if (old_fld == 4 && cur_fld > 4)
{
    if (osub_date (osub_buf.osub_date,16,38)==FALSE)
    {
        cur_fld = old_fld ;
        *st = RETURN ;
        return ;
    }
    return ;
}
}
```



```

#include "extern.inc"
#include "stue.inc"

int    total ;

conclude_reg ()
{
    int    func_no ;

    mon_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,moh_var,mon_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
    if (STATUS==0)
    {
        mon_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
        mon_clr0 () ;
        mon_print () ;
    }
    do {
        switch (func_no = mon_menu())
        {
            case F1:
                mon_ins () ;
                break ;
            case F2:
                mon_del () ;
                break ;
            case F3:
                mon_oth () ;
                break ;
        }
    } while (func_no != F9) ;
}

mon_rec2buf ()
{
    int    i ;

    strcpy (mon_buf.id,mon_rec.id) ;
    mon_buf.depp = mon_rec.depp ;
    mon_buf.sem = mon_rec.sem ;
    mon_buf.year = mon_rec.year ;
    for (i=0; i<12; i++)
    {
        strcpy (mon_buf.sub[i].code,mon_rec.sub[i].code) ;
    }
}

mon_buf2rec ()
{
    int    i ;

    strcpy (mon_rec.id,mon_buf.id) ;
    mon_rec.depp = mon_buf.depp ;

```

```

mon_rec.sem = mon_buf.sem ;
mon_rec.year = mon_buf.year ;
for (i=0; i<12; i++)
{
    mon_buf.sub[i].code[6] = '\0' ;
    strcpy (mon_rec.sub[i].code,mon_buf.sub[i].code) ;
}

mon_scr ()
{
    clrscr () ;
    frame ( 2, 1,20,79,"ระบบบงกชของมหาวิทยาลัยเกษตรศาสตร์") ;
    lineH ( 1,79, 7) ;
    lineV ( 7,20,10) ;
    lineV ( 7,20,69) ;
    printp ( 4,13,"ช.อ. :",NM_VDO) ;
    printp ( 5, 9,"ภาควิชา :",NM_VDO) ;
    printp ( 7, 3,"รหัสวิชา",RV_VDO) ;
    printp ( 7,32,"ช.อ.วิชา",RV_VDO) ;
    printp ( 7,71,"หน่วยกิต",RV_VDO) ;
    mon_clr0();
    printp (21,66,"รวม :",NM_VDO) ;
    printp (21,72,"0",RV_VDO) ;
}

mon_menu ()
{
    static struct lbl_template fn_mon[] = {
        {23, 4,"1:เพิ่มข้อมูล",RV_VDO},
        {23,23,"2:ถอนข้อมูล",RV_VDO},
        {23,38,"3:นำข้อมูลออก",RV_VDO},
        {23,71,"9:Exit",RV_VDO}
    };

    int i ;

    i = get_func (F3,fn_mon) ;
    return (i) ;
}

mon_print ()
{
    int i ;
    mon_name();
    mon_dep();
    mon_sem();
    total = 0 ;
    for (i=0; i<12; i++)
    {
        if (mon_buf.sub[i].code[0] != '\0')
            mon_getsub (i) ;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        mon_total () ;
    }

mon_name ()
{
    int    i ;

    strcpy (stu_key0.id,mon_buf.id) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบรหัสนักเรียนในแฟ้มข้อมูล") ;
        return (FALSE) ;
    }
    printp ( 4,18,"                                ",NM_VDO)
;
    printp ( 4,18,stu_rec.th_name,NM_HL_VDO) ;
    i = 0 ;
    while (stu_rec.th_name[i] != ' ')
        i++ ;
    printp ( 4,20+i,stu_rec.th_sname,NM_HL_VDO) ;
    return (TRUE) ;
}

mon_total ()
{
    char    tbuf[3] ;

    printp (21,72," ",NM_VDO) ;
    itoa_1 (total,tbuf,2) ;
    printp (21,72,tbuf,NM_HL_VDO) ;
}

mon_clr0 ()
{
    int    i ;

    for (i=0; i<12; i++)
    {
        printp (8+i,3," ",NM_VDO) ;
        printp (8+i,12,"
",NM_VDO) ;
        printp (8+i,71," ",NM_VDO) ;
    }
}

mon_getsub (int i)
{
    char    gbuf[2] ;
    char    gbuf1[61] ;
    int     n ;

    mon_buf.sub[i].code[6] = '\0' ;
    strcpy (osub_key0.osub_id,mon_buf.sub[i].code) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;

```

```

if (STATUS != 0)
{
    b_error ("ไม่พบการเปิดสอนกระบวนวิชา");
    return (FALSE);
}
strcpy (subj_key0.subj_code,osub_rec.osub_id);
STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
if (STATUS != 0)
{
    b_error ("ไม่พบกระบวนวิชาในแฟ้มข้อมูล");
    return (FALSE);
}
strcpy (gbuf1,subj_rec.subj_name);
n = 0;
while (gbuf1[n] != ' ' || gbuf1[n+1] != ' ')
    n++;
gbuf1[n] = '\0';
printf (8+i,3,subj_rec.subj_code,NM_VDO);
printf (8+i,14,"",NM_VDO);

printf (8+i,14,gbuf1,NM_HL_VDO);
itoa_1 (subj_rec.subj_cred,gbuf,1);
printf (8+i,73,gbuf,NM_HL_VDO);
total += subj_rec.subj_cred;
return (TRUE);
}

depp(c,b)
int c;
char b[];
{
    static char *dep[] = {"คณะศึกษาศาสตร์และวิทยาการคอมพิวเตอร์",
        "เคมีอุตสาหกรรม",
        "เทคโนโลยีชีวภาพ",
        "ฟิสิกส์ประยุกต์",
        "สถิติประยุกต์"};

    strcpy(b,dep[c]);
}

mon_dep ()
{
    char part2[41];
    depp(mon_buf.depp,part2);
    printf(mon_var[1].row,mon_var[1].col,part2,NM_HL_VDO);
    printf (5,18,"",NM_VDO);
    printf (5,18,part2,NM_HL_VDO);
    return (TRUE);
}

semm(cd,bb)
int cd;
char bb[];
{
    static char *sem[] = {"1st",
        "2nd",

```

```

        "Summer" ] ;
    strcpy(bb,sem[cd]);
}
mon_sem ()
{
    char    sbuf[7] ;
    semm(mon_buf.sem,sbuf);
    printp(mon_var[2].row,mon_var[2].col,sbuf,NM_HL_VDO);
    printp(6,18,"          ",NM_VDO);
    printp( 6,18,sbuf,NM_HL_VDO) ;
    return(TRUE);
}

mon_year ()
{
    char    ybuf[2] ;

    printp ( 6,38," ",NM_VDO) ;
    itoa_1 (mon_buf.year,ybuf,1) ;
    printp ( 6,38,ybuf,NM_HL_VDO) ;
}

mon_ins ()
{
    static struct var_template    cins_var[] = {
        {19,40,STRING,"99-999","99-999",}
    } ;

    static struct lbl_template    cins_lbl[] = {
        {19,31,"รพ.สวนป่า :",NM_VDO}
    } ;

    int     status, dflag = -1 ;
    char    dbuf[7], gbuf[2] ;
    int     subptr = 0,i ;

    if (mon_buf.id[0] == ' ')
    {
        error ("ต้องใส่ รพ.ต้นปลูกจาก เอน") ;
        return ;
    }
    cins_var[0].datatag.str_add = dbuf ;
    make_win (16,29,21,50," ") ;
    printp (18,30,"ป.บรพ.สวนป่า คณะกรรมการพ.ม",NM_VDO) ;
    scr_mgr (NEW,DISP_FLAG,NEW,cins_var,cins_lbl,1,NULL,NULL) ;
    T_E_flag = Eng ;
    cur_fld = 0;

    do {
        status = scr_mgr (OLD,ED_FLAG,OLD,cins_var,cins_lbl,1,NULL,

```



```

    } while (status != F10 && status != ESC && status != RETURN) ;
recall () ;
if (status == ESC)
{
    return ;
}
dbuf[6] = '\0' ;
for (i=0; i<12; i++)
{
    if (mon_buf.sub[i].code[0] != ' ')
        subptr++ ;
    if ( (strcmp (mon_buf.sub[i].code,dbuf)==0))
        dflag = i ;
}

if (subptr == 12 || total > 23 )
{
    error ("ไม่สามารถเพิ่มรายการลงทะเบียนได้อีก") ;
    strcpy(mon_buf.sub[subptr].code,"");
    return ;
}

if (dflag != -1)
{
    error ("วิชาลงทะเบียนไปแล้ว") ;
    return ;
}
dflag = subptr +1;
strcpy (mon_buf.sub[dflag].code,dbuf) ;
mon_sortsub () ;
mon_buf2rec () ;
STATUS = BTRV (B_UPDATE,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
if (STATUS != 0)
    b_error ("ไม่สามารถแก้ไขข้อมูล") ;
scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
mon_clr0 () ;
mon_print () ;

}

mon_del ()
{
    static struct var_template    cdel_var[] = {
        {19,40,STRING,"99-999","99-999",}
    } ;

    static struct lbl_template    cdel_lbl[] = {
        {19,31,"รหัสวิชา :",NM_VDO}
    } ;

    int    status, i, dflag ;
    char    dbuf[7] ;

    if (mon_buf.id[0] == ' ')

```

```

        return ;
    }
    cdel_var[0].datatag.str_add = dbuf ;
    make_win (16,29,21,50," ") ;
    printp (18,31,"โปรแกรมตรวจสอบการลบ",NM_VDO) ;
    scr_mgr (NEW,DISP_FLAG,NEW,cdel_var,cdel_lbl,1,NULL,NULL) ;
    T_E_flag = Eng ;
    cur_fld = 0 ;
    do {
        status = scr_mgr (OLD,ED_FLAG,OLD,cdel_var,cdel_lbl,1,NULL,
NULL) ;
    } while (status != F10 && status != ESC && status != RETURN) ;
    recall () ;
    if (status == ESC)
        return ;
    dflag = -1 ;
    dbuf[6] = '\0' ;
    for (i=0; i<12; i++)
    {
        if (strcmp (mon_buf.sub[i].code,dbuf)==0)
            dflag = i ;
    }
    if (dflag == -1)
    {
        error ("ยังไม่ได้ลงทะเบียนรหัส") ;
        return ;
    }
    strcpy (subj_key0.subj_code,mon_buf.sub[dflag].code) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    total -= subj_rec.subj_cred ;
    strcpy (mon_buf.sub[dflag].code,"") ;
    mon_sortsub () ;
    mon_buf2rec () ;
    STATUS = BTRV (B_UPDATE,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
    if (STATUS != 0)
        b_error ("ไม่สามารถแก้ไขข้อมูลได้") ;
    scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
    mon_clr0 () ;
    mon_print () ;
}

mon_oth ()
{
    int      status, st, i ;

    scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
    clr_win (5,18,5,50) ;
    clr_win (6,18,6,22) ;
    clr_win (8,11,19,68) ; clr_win (8,70,19,78) ;
    printp (4,18,"",NM_VDO) ;
}

```

เอกสารนี้เป็นเอกสารที่ `mon_clr0()`; การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
printp(21,58," ",NM_VDO);
```

ไม่ว่ากรณีใดๆทั้งสิ้น ไม่สามารถนำเอกสารนี้ไปดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
T_E flag = Eng;
```

```

cur_fld = 0 ;
do {
    if (cur_fld==1)
    {
        T_E_flag = Eng;
        status = fld_sel (&mddep_var[0],&dep_selv[0],dep_sel
v[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            mon_buf.depp = dep_selv[0].cur_sv ;
            cur_fld = 2 ;
    }
    if (cur_fld==2)
    {
        T_E_flag = Eng;
        status = fld_sel (&mplsem_var[0],&ssem_selv[0],ssem
_selv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            mon_buf.sem = ssem_selv[0].cur_sv ;
        if (status==CUR_UP)
            cur_fld = 1 ;
        else
            cur_fld = 3 ;
    }
    status = scr_mgr (OLD,ED_FLAG,OLD,mon_var,mon_lbl,4,NULL,NUL
L) ;
    mon_fld(&status);
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    STATUS = BTRV (B_GETPR,mon_blk,&mon_rec,&mon_len,&mon_key0,
0) ;
    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,mon_blk,&mon_rec,&mon_len,&mon_ke
y0,0) ;

    scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
    if (STATUS == 0)
    {
        mon_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,N
ULL) ;

        mon_print () ;
    }
    return ;
}
plan_key0.plan_dep = mon_buf.depp;
plan_key0.plan_sem = mon_buf.sem;
plan_key0.plan_year = mon_buf.year ;
STATUS = BTRV (B_GETEQ,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
if (STATUS==0)
{
    mon_buf.depp= plan_rec.plan_dep;
    mon_buf.sem = plan_rec.plan_sem ;
    mon_buf.year = plan_rec.plan_year ;
    for (i=0; i<12; i++)
        strcpy (mon_buf.sub[i].code,plan_rec.sub[i].code) ;
}

```

```

mon_buf2rec ( ) ;
STATUS = BTRV (B_INSERT,mon_blk,&mon_rec,&mon_len,&mon_key0
,0) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถเก็บข้อมูลได้") ;
    STATUS = BTRV (B_GETPR,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,mon_blk,&mon_rec,&mon_len,&mon_key0,0) ;
    scr_mgr (NEW,DISP_FLAG,NEW,mon_var,mon_lbl,4,NULL,NULL) ;
    printp (4,18,"
    ",NM_VDO) ;
    if (STATUS == 0)
    {
        mon_rec2buf ( ) ;
        scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
        mon_print ( ) ;
    }
    return ;
}
scr_mgr (OLD,DISP_FLAG,OLD,mon_var,mon_lbl,4,NULL,NULL) ;
mon_clr0 ( ) ;
mon_print ( ) ;
}
else
    error ("do nothing") ;
}

```

```

mon_sortsub (

```

```

{
    char    scode[12][6] ;
    int     tmptmp ;
    int     point1, point2 ;
    int     tmptag[12] ;

    for (point1=0; point1<12; point1++)
    {
        strcpy (scode[point1],mon_buf.sub[point1].code) ;
        tmptag[point1] = point1 ;
    }
    for (point1=0; point1<11; point1++)
    {
        for (point2=point1+1; point2<12; point2++)
        {
            if (strcmp(scode[tmptag[point1]],scode[tmptag[point
2]]) > 0)
            {
                tmptmp = tmptag[point1] ;
                tmptag[point1] = tmptag[point2] ;
                tmptag[point2] = tmptmp ;
            }
        }
    }
}

```

```

        tmptag[point1] = tmptag[point2] ;
        tmptag[point2] = tmptmp ;
    }
}
point1 = 0 ;
while (scode[tmptag[point1]][0] == ' ')
    point1++ ;
while (point1 < 12)
{
    tmptmp = tmptag[11] ;
    for (point2=10; point2 >= 0; point2--)
        tmptag[point2+1] = tmptag[point2] ;
    tmptag[0] = tmptmp ;
    point1++ ;
}
for (point1=0; point1<12; point1++)
    strcpy (mon_buf.sub[point1].code,scode[tmptag[point1]]);
}

mon_fld (stat)
int *stat;
{
    if (*stat == ESC)
        return ;
    if (old_fld ==0 || *stat == F10 )
    {
        if ((mon_name()) ==FALSE)
        {
            *stat = RETURN;
            return;
        }
    }

    if (old_fld ==1 || *stat == F10 )
    {
        if (mon_buf.depp > 4 )
        {
            error("โปรดระบุภาควิชา");
            cur_fld = old_fld;
            *stat = RETURN ;
            return;
        }
    }

    if (old_fld ==2 || *stat == F10)
    {
        if (mon_buf.sem > 2 )
        {
            error ("โปรดระบุเพศ") ;
            cur_fld = old_fld ;
            *stat = RETURN ;
            return;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include "extern.inc"
```

```
#include "stue.inc"
```

```
student ()
```

```
{
```

```
    int    func_no ;
```

```
    stu_scr () ;
```

```
    scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,NULL) ;
```

```
    STATUS = BTRV (B_GETLW,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
```

```
    if (STATUS == 0)
```

```
    {
```

```
        cpy_stu () ;
```

```
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
```

```
        stu_print () ;
```

```
        stu_datel () ;
```

```
        stu_date2 () ;
```

```
    }
```

```
    do
```

```
    {
```

```
        switch (func_no=stu_menu())
```

```
        {
```

```
            case F1 :
```

```
                stu_prev () ;
```

```
                break ;
```

```
            case F2 :
```

```
                stu_next () ;
```

```
                break ;
```

```
            case F3 :
```

```
                stu_srch () ;
```

```
                break ;
```

```
            case F4 :
```

```
                stu_edit () ;
```

```
                break ;
```

```
            case F5 :
```

```
                stu_ins () ;
```

```
                break ;
```

```
            case F6 :
```

```
                stu_del () ;
```

```
                break ;
```

```
        }
```

```
    } while (func_no != F9) ;
```

```
}
```

```
stu_cpy ()
```

```
{
```

```
    strcpy (stu_rec.stu_id,stu_buf.stu_id) ;
```

```
    stu_rec.part = stu_buf.part;
```

```
    strcpy (stu_rec.th_name,stu_buf.th_name) ;
```

```
    stu_rec.sex = stu_buf.sex ;
```

```
    strcpy (stu_rec.th_sname,stu_buf.th_sname) ;
```

```
    strcpy (stu_rec.ename,stu_buf.ename) ;
```

```
    strcpy (stu_rec.e_sname,stu_buf.e_sname) ;
```

```
    strcpy (stu_rec.d_birth,stu_buf.d_birth) ;
```

```
    strcpy (stu_rec.d_cont,stu_buf.d_cont) ;
```



```

strcpy (stu_rec.adv_name,stu_buf.adv_name);
strcpy (stu_rec.add1,stu_buf.add1) ;
strcpy (stu_rec.add2,stu_buf.add2) ;
strcpy (stu_rec.tel,stu_buf.tel) ;
strcpy (stu_rec.p_name,stu_buf.p_name) ;
strcpy (stu_rec.p_sname,stu_buf.p_sname) ;
strcpy (stu_rec.p_add1,stu_buf.p_add1) ;
strcpy (stu_rec.p_add2,stu_buf.p_add2) ;
strcpy (stu_rec.p_tel,stu_buf.p_tel) ;
}

```

```
stu_cpy1 ()
```

```

{
    stu_rec.stu_ocrad = 0 ;
    stu_rec.stu_osco = 0.0 ;
    stu_rec.stu_ncrad = 0 ;
    stu_rec.stu_nsco = 0.0 ;
    stu_rec.stu_crad = 0 ;
    stu_rec.stu_sco = 0.0 ;
}

```

```
cpy_stu ()
```

```

{
    strcpy (stu_buf.stu_id,stu_rec.stu_id) ;
    stu_buf.part = stu_rec.part;
    strcpy (stu_buf.th_name,stu_rec.th_name) ;
    stu_buf.sex = stu_rec.sex ;
    strcpy (stu_buf.th_sname,stu_rec.th_sname) ;
    strcpy (stu_buf.ename,stu_rec.ename) ;
    strcpy (stu_buf.e_sname,stu_rec.e_sname) ;
    strcpy (stu_buf.d_birth,stu_rec.d_birth) ;
    strcpy (stu_buf.d_cont,stu_rec.d_cont) ;
    strcpy (stu_buf.adv_name,stu_rec.adv_name);
    strcpy (stu_buf.add1,stu_rec.add1) ;
    strcpy (stu_buf.add2,stu_rec.add2) ;
    strcpy (stu_buf.tel,stu_rec.tel);
    strcpy (stu_buf.p_name,stu_rec.p_name) ;
    strcpy (stu_buf.p_sname,stu_rec.p_sname) ;
    strcpy (stu_buf.p_add1,stu_rec.p_add1) ;
    strcpy (stu_buf.p_add2,stu_rec.p_add2) ;
    strcpy (stu_buf.p_tel,stu_rec.p_tel) ;
}

```

```
stu_scr ()
```

```

{
    clrscr () ;
    frame ( 1, 2,21,80,"ประมวลผลงานทะเบียนนักศึกษา" ) ;
    printp (9,39,"[dd/mm/yy]",NM_VDO) ;
    printp (10,39,"[dd/mm/yy]",NM_VDO) ;
    printp (19,39,"หน่วยกิตสะสม :",NM_VDO) ;
}

```

```
stu_menu ()
```

```
int i;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

static struct lbl_template fu_stu[] = {
    {23, 4, "1:ก.ณ", RV_VDO},
    {23, 11, "2:พล.", RV_VDO},
    {23, 18, "3:ค.นท.", RV_VDO},
    {23, 26, "4:นก.ช", RV_VDO},
    {23, 35, "5:เพ.ม.เด.ม", RV_VDO},
    {23, 46, "6:ลบ", RV_VDO},
    {23, 71, "9:Exit", RV_VDO}
},

i = get_func (F6, fu_stu) ;
return (i) ;
}

stu_fld (stat)
int *stat ;
{
    if (*stat == ESC)
        return ;
    if (old_fld == 1 || *stat == F10 )
    {
        if (stu_buf.part > 4 )
        {
            error("โปรแกรมทศ.");
            cur_fld = old_fld;
            *stat = RETURN ;
            return;
        }
    }
    if (old_fld == 3 || *stat == F10)
    {
        if (stu_buf.sex > 1)
        {
            error ("โปรแกรมเพศ");
            cur_fld = old_fld ;
            *stat = RETURN ;
            return;
        }
    }
    if (old_fld == 7 || *stat == F10 )
    {
        if (chk_date(stu_var[7].datatag.str_add) == FALSE)
        {
            printf ( 9, 53, " ", NM_VDO)

            error ("ว.นท. ผลผลิต..");
            cur_fld = old_fld ;
            *stat = RETURN ;
        }
        else{
            stu_datel () ;
        }
    }
    return;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (old_fld==8 ;; *stat == F10 )
{
    if (chk_date(stu_var[8].datatag.str_add)==FALSE)
    {
        printp ( 10,53,"                                ",NM_VDO).

;

        error ("ว.พท.ผิดพลาด..") ;
        cur_fld = old_fld ;
        *stat = RETURN ;
    }
    else{
        stu_date2 ( ) ;
    }
}
return;
}

stu_date1 (
{
    char    buf[25] ;

    if (chk_date(stu_var[7].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (stu_var[7].datatag.str_add,buf) ;
    printp ( 9,53,"                                ",NM_VDO) ;
    printp ( 9,53,buf,NM_HL_VDO) ;
}

stu_date2 (
{
    char    buf[25];

    if (chk_date(stu_var[8].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (stu_var[8].datatag.str_add,buf) ;
    printp ( 10,53,"                                ",NM_VDO) ;
    printp ( 10,53,buf,NM_HL_VDO) ;
}

stu_dep(cod,buff)
int cod;
char buff[];
{
    static char *dep [] = { "คณบดีศาสตรและวิทยาการคอมพิวเตอร์", "เคม๒",
        "ชีววิทยาประย กต๑", "ฟ๑ส๑กล๑ส๑ประย กต๑",
        "ส๑ก๑ด๑ประย กต๑" };

    strcpy(buff,dep[cod]);
}

stu_print (
{
    char    buf1[41] ,buf2[5] ,buf3[3] ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับเอกรใช้งานเพื่อการสืบพวเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    stu_dep (stu_buf.part,buf1);
    printp (stu_var[1].row,stu_var[1].col,buf1,NM_HL_VDO) ;
    printp (4,20,"                                ",NM_HL_VDO) ;
;
    printp (4,20,buf1,NM_HL_VDO);
    stu_sex (stu_buf.sex,buf2) ;
    printp (stu_var[3].row,stu_var[3].col,buf2,NM_HL_VDO) ;
    printp (5,55,"          ",NM_HL_VDO);
    printp (5,55,buf2,NM_HL_VDO);
    printp (19,52,"          ",NM_HL_VDO);
    itoa_1 (stu_rec.stu_crad,buf3,3) ;
    printp (19,52,buf3,NM_HL_VDO) ;
}

stu_sex (code,buf)
int    code ;
char   buf[] ;
{
    static char *sex[] = {"ชาย",
                           "หญิง"} ;

    strcpy (buf,sex[code]) ;
}

stu_prev ()
{
    STATUS = BTRV (B_GETPR,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS == 0)
    {
        cpy_stu () ;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stu_print () ;
        stu_datel () ;
        stu_date2 () ;
    }
    else
        b_error ("กดขงมองผลแรกแล้วครับ") ;
    return ;
}

stu_next ()
{
    STATUS = BTRV (B_GETNX,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS == 0)
    {
        cpy_stu () ;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stu_print () ;
        stu_datel () ;
        stu_date2 () ;
    }
    else
        b_error ("กดขงมองผลสุดท้ายแล้วครับ") ;
    return ;
}

```

}

stu_srch ()

{

int row,col,status,new;

char buff[10];

T_E_flag = Eng ;

cur_fld = 0 ;

make_win (18, 6,22,36,"Search Window") ;

printw(20,9,"รหัสนักศึกษา",NM_HL_VDO);

strcpy(buff,stu_buf.stu_id);

row = stu_var[0].row;

col = stu_var[0].col;

stu_var[0].row = 20;

stu_var[0].col = 21;

new = NEW;

do {

status = fld_edit(new,&stu_var[0],NAUTO_ADV,NULL,NULL) ;

new = OLD;

} while (status != RETURN && status != F10 && status !=ESC);

stu_var[0].row = row ;

stu_var[0].col = col;

recall();

if (status == ESC)

{

strcpy(stu_buf.stu_id,buff);

return;

}

strcpy(stu_key0.id,stu_buf.stu_id);

STATUS = BTRV(B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0);

if (STATUS == 0)

{

cpy_stu () ;

scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;

stu_print () ;

stu_datel () ;

stu_date2 () ;

return ;

}

else

b_error ("ไม่พบรหัสนักศึกษานี้ในแฟ้มของมจร") ;

return ;

}

stu_edit ().

{

int status,sure,STAT1 ;

cur_fld = 0 ;

T_E_flag = Eng ;

do {

if (cur_fld != 1)

T_E_flag = Eng ;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        status = fld_sel (&ssdep_var[0],&dep_selv[0],dep_se
lv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            stu_buf.part = dep_selv[0].cur_sv ;
        cur_fld = 2 ;
    }
    if (cur_fld == 3)
    {
        T_E_flag = Eng ;
        status = fld_sel (&sssex_var[0],&sex_selv[0],sex_se
lv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            stu_buf.sex = sex_selv[0].cur_sv ;
        cur_fld = 4 ;
    }
    switch (cur_fld)
    {
        case 2 :
        case 4 :
        case 9 :
        case 10 :
        case 11 :
        case 13 :
        case 14 :
        case 15 :
        case 16 :
            T_E_flag = Thai ;
            break ;
        default :
            T_E_flag = Eng ;
            break ;
    }
    status = scr_mgr (OLD,ED_FLAG,OLD,stu_var,stu_lbl,18,NULL,N
ULL) ;
    stu_fld (&status) ;
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    cpy_stu () ;
    scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
    stu_print () ;
    stu_datel () ;
    stu_date2 () ;
    STATUS=BTRV(B_UNLOCK,stu_blk,&stu_rec,&stu_len,&stu_key0,0)
;

    return ;
}
sure = make_sure ("โปรดขออนุญาตในการแก้ไขข้อมูล") ;
if (sure == 0)
    return ;
stu_cpy () ;
instr (0,"การแก้ไขข้อมูล...") ;
STATUS = BTRV (B_UPDATE,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
if (STATUS != 0)

```

เอกสารนี้เป็นเอกสารที่... ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

b_error ("ไม่สามารถเก็บข้อมูลได้");
STATUS = BTRV (B_GETGE,stu_blk,&stu_rec,&stu_len,&stu_key0,
0);
if (STATUS != 0)
scr_mgr (NEW,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,
NULL) ;
else
{
cpy_stu () ;
scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,
NULL) ;

stu_print () ;
stu_date1 () ;
stu_date2 () ;
}
return ;
}
scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
stu_print () ;
stu_date1 () ;
stu_date2 () ;
}
stu_ins()
{
int status, sure, STAT1 ;

cur_fld = 0;
scr_mgr(NEW,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL);
printp(4,20,"",NM_VDO);
printp(5,55,"",NM_VDO);
printp(9,53,"",NM_VDO);
printp(10,53,"",NM_VDO);
printp(19,52,"",NM_VDO) ;
T_E_flag = Eng;
do{
if (cur_fld == 1)
{
T_E_flag = Eng ;
status = fld_sel (&ssdep_var[0],&dep_selv[0],dep_selv[0].svar_struct,NULL,NULL) ;
if (status != ESC)
stu_buf.part = dep_selv[0].cur_sv ;
cur_fld = 2 ;
}
if (cur_fld == 3)
{
T_E_flag = Eng ;
status = fld_sel (&sssex_var[0],&sex_selv[0],sex_selv[0].svar_struct,NULL,NULL) ;
if (status != ESC)
stu_buf.sex = sex_selv[0].cur_sv ;
cur_fld = 4 ;
}
}
switch (cur_fld)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    case 2 :
    case 4 :
    case 9 :
    case 10 :
    case 11 :
    case 13 :
    case 14 :
    case 15 :
    case 16 :
        T_E_flag = Thai ;
        break ;
    default :
        T_E_flag = Eng ;
        break ;
}

status = scr_mgr (OLD,ED_FLAG,OLD,stu_var,stu_lbl,18,NULL,N
ULL) ;

    stu_fld (&status) ;
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    cpy_stu () ;
    scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
    stu_print () ;
    stu_date1 () ;
    stu_date2 () ;
    return ;
}
sure = make_sure ("โปรดอย่าขโมยงานของเรา...เมื่อก่อนขโมยมา") ;
if (sure == 0)
    return ;
stu_cpy () ;
stu_cpy1 () ;
STATUS = BTRV (B_INSERT,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถเก็บข้อมูลเข้าแฟ้มได้") ;
    STATUS = BTRV (B_GETPR,stu_blk,&stu_rec,&stu_len,&stu_key0,
0) ;

    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,stu_blk,&stu_rec,&stu_len,&stu_key0,
0) ;

    if (STATUS != 0)
        STATUS = BTRV (B_GETLW,stu_blk,&stu_rec,&stu_len,&stu_key0,
0) ;

    if (STATUS != 0)
    {
        scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,
NULL) ;

        return ;
    }
}

scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
stu_print () ;

```

```

        stu_date1 () ;
        stu_date2 () ;
    }

stu_del ()
{
    int    sure ;

    sure = make_sure ("โปรดขออนุญาตลบข้อมูล") ;
    if (sure==0)
        return ;
    /*STATUS=BTRV(B_GETEQL,stu_blk,&stu_rec,&stu_len,&stu_key0,0); */
    instr (0,"การลบข้อมูล....") ;
    STATUS = BTRV (B_DELETE,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    clrinstr () ;
    if (STATUS == 0)
    {
        scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,NULL) ;
        STATUS = BTRV(B_UNLOCK,stu_blk,&stu_rec,&stu_len,&stu_key0,
0);
        STATUS = BTRV (B_GETNX,stu_blk,&stu_rec,&stu_len,&stu_key0,
0) ;
        if (STATUS != 0)
        STATUS = BTRV (B_GETPR,stu_blk,&stu_rec,&stu_len,&stu_key0,
0) ;
        if (STATUS != 0)
        {
            scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,
NULL) ;
            printp (4,20,"
",NM_HL_VDO);
            printp (5,55,"          ",NM_HL_VDO);
            printp(9,53,"          ",NM_VDO);
            printp(10,53;"          ",NM_VDO);
            return;
        }
        cpy_stu () ;
        STATUS=BTRV(B_UNLOCK,stu_blk,&stu_rec,&stu_len,&stu_key0,0)
;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stu_print () ;
        stu_date1 () ;
        stu_date2 () ;
        return ;
    }
    b_error ("ไม่สามารถลบข้อมูลได้") ;
    scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
    stu_print () ;
    stu_date1 () ;
    stu_date2 () ;
}

```

```

#include "extern.inc"
#include "stue.inc"

gen_sub ()
{
    int    func_no ;

    gens_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,gens_var,gens_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;
    if (STATUS ==0)
    {
        gens_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
        gens_print () ;
    }
    gens_clr0 () ;
    do
    {
        switch (func_no= gens_menu())
        {
            case F1 :
                gens_prev () ;
                break;
            case F2 :
                gens_next () ;
                break;
            case F3 :
                gens_edit () ;
                break;
            case F4 :
                gens_ins () ;
                break;
            case F5 :
                gens_del () ;
                break;
        }
    } while (func_no != F9) ;
}

gens_scr ()
{
    clrscr () ;
    frame (3,1,12,79,"หมวดข่าวฉบับ") ;
    lineH (1,79,6) ;
    printp (8,12,"ชื่อนาม : ",NM_VDO) ;
    printp (9,11,"นามสกุล : ",NM_VDO) ;
    printp (10,13,"อาชีพ : ",NM_VDO) ;
    printp (10,27,"ปีเกิด : ",NM_VDO) ;
}

gens_rec2buf ()
{
    gens_buf.gens_dep = gens_rec.gens_dep ;
    gens_buf.gens_year = gens_rec.gens_year ;
}

```

```

gens_buf.gens_sem = gens_rec.gens_sem ;
strcpy (gens_buf.code,gens_rec.code) ;
}

gens_buf2rec ()
{
    gens_rec.gens_dep = gens_buf.gens_dep ;
    gens_rec.gens_year = gens_buf.gens_year ;
    gens_rec.gens_sem = gens_buf.gens_sem ;
    strcpy (gens_rec.code,gens_buf.code) ;
}

gens_print ()
{
    gens_dep () ;
    gens_sem () ;
    gens_getsub () ;
}

gens_dep ()
{
    static char *dep[] = {"คณะศาสตร์และวิทยาการคอมพิวเตอร์",
        "เคมีอุตสาหกรรม",
        "เทคโนโลยีชีวภาพ",
        "ฟิสิกส์ประยุกต์",
        "สถิติประยุกต์"} ;

    char dbuf[41] ;

    strcpy (dbuf,dep[gens_buf.gens_dep]) ;
    printp (4,20,"",NM_VDO) ;
    printp (4,20,dbuf,NM_HL_VDO) ;
}

gens_sem ()
{
    static char *sem[] = {"1st",
        "2nd",
        "Summer"} ;

    char sbuf[7] ;

    strcpy (sbuf,sem[gens_buf.gens_sem]) ;
    printp (5,41,"",NM_VDO) ;
    printp (5,41,sbuf,NM_HL_VDO) ;
}

gens_getsub ()
{
    char cbuf[2], tbuf[2], obuf[2] ;
    char gbuf[61] ;
    int n ;

    strcpy (osub_key0.osub_id,gens_buf.code) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)

```

```

        b_error("ไม่มีการเปิดสอนกระบวนวิชา");
        return (FALSE);
    }
    strcpy (subj_key0.subj_code,osub_rec.osub_id);
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if (STATUS != 0)
    {
        b_error ("ไม่พบกระบวนวิชาในแฟ้มข้อมูล");
        return (FALSE);
    }
    strcpy (gbuf,subj_rec.subj_name);
    n = 0;
    while (gbuf[n] != ' ' || gbuf[n+1] != ' ')
        n++;
    gbuf[n] = '\0';
    printp (8,20,"
NM_VDO);
    printp (8,20,gbuf,NM_HL_VDO);
    itoa_1 (subj_rec.subj_cred,cbuf,1);
    printp (9,20,cbuf,NM_HL_VDO);
    itoa_1 (subj_rec.subj_theory,tbuf,1);
    printp (10,20,tbuf,NM_HL_VDO);
    itoa_1 (subj_rec.subj_opr,obuf,1);
    printp (10,34,obuf,NM_HL_VDO);
    return (TRUE);
}

gens_clr0 ()
{
    if (gens_buf.code[0] == ' ')
    {
        printp (8,20,"
,NM_VDO);
        clr_win (9,20,10,22);
        clr_win (10,34,10,50);
    }
}

gens_menu ()
{
    static struct lbl_template fn_gens[] = {{23, 4,"1:ก.ณ",RV_VDO},
                                             {23,11,"2:ท.ง",RV_VDO},
                                             {23,18,"3:แ.ง",RV_VDO},
                                             {23,26,"4:พ.ม.ด.ม",RV_VDO},
                                             {23,35,"5:ลป",RV_VDO},
                                             {23,71,"9:Exit",RV_VDO}
    };

    int i;

    i = get_func (F5,fn_gens);
    return (i);
}

gens_prev ()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

STATUS = BTRV (B_GETPR,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;
if (STATUS != 0)
{
    b_error ("การประมวลผลรายการผิดพลาด") ;
    return ;
}
gens_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
gens_print () ;
gens_clr0 () ;
}

gens_next ()
{
    STATUS = BTRV (B_GETNX,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("การประมวลผลรายการผิดพลาด") ;
        return ;
    }
    gens_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
    gens_print () ;
    gens_clr0 () ;
}

gens_edit ()
{
    int    status, st ;

    T_E_flag = Eng ;
    cur_fld = 0 ;
    do
    {
        if (cur_fld == 0)
        {
            status = fld_sel (&gsdep_var[0],&dep_selv[0],dep_selv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                gens_buf.gens_dep = dep_selv[0].cur_sv ;

            if (status==F10 || status==ESC)
                break ;
            cur_fld = 1 ;
        }
        if (cur_fld ==2)
        {
            status = fld_sel (&gssem_var[0],&ssem_selv[0],ssem_selv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                gens_buf.gens_sem = ssem_selv[0].cur_sv ;
            if (status == ESC || status == F10)
                break ;
            if (status==CUR_UP)
                cur_fld = 1 ;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        else
            cur_fld = 3 ;
        }
        if (cur_fld != 0)
            status = scr_mgr (OLD,ED_FLAG,OLD,gens_var,gens_lbl
,4,NULL,NULL) ;
        if (old_fld == 3)
        {
            st = gens_getsub() ;
            if (st == FALSE)
                cur_fld = 3;
        }
        . } while (status != F10 && status != ESC) ;
        if (status == ESC)
        {
            STATUS = BTRV (B_GETEQ,gens_blk,&gens_rec,&gens_len,&gens_k
ey0,0) ;
            gens_rec2buf () ;
            scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
            gens_print () ;
            gens_clr0 () ;
            return ;
        }
        instr (0,"การแก้ไขข้อมูล...") ;
        gens_buf2rec () ;
        STATUS = BTRV (B_UPDATE,gens_blk,&gens_rec,&gens_len,&gens_key0,0)
;
        if (STATUS != 0)
        {
            b_error ("ไม่สามารถแก้ไขแฟ้มข้อมูลได้") ;
            STATUS = BTRV (B_GETGE,gens_blk,&gens_rec,&gens_len,&gens_k
ey0,0) ;
            if (STATUS != 0)
                scr_mgr (NEW,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL
,NULL) ;
            else
            {
                gens_rec2buf () ;
                scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL
,NULL) ;

                gens_print () ;
                gens_clr0 () ;
            }
            return ;
        }
        scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
        gens_print () ;
        gens_clr0 () ;
    }

    gens_ins ()
    {
        int      status, st, STAT1 ;

        scr_mgr (NEW,DISP_FLAG,NEW,gens_var,gens_lbl,4,NULL,NULL) ;

```

```

printp (4,20,"
printp (5,41,"      ",NM_VDO) ;
clr_win (8,20,9,70) ;
clr_win (10,20,10,26) ;
clr_win (10,34,10,70) ;
T_E_flag = Eng ;
cur_fld = 0 ;
do {
    if (cur_fld == 0)
    {
        status = fld_sel (&gsdep_var[0],&dep_selv[0],dep_se
lv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            gens_buf.gens_dep = dep_selv[0].cur_sv ;
        if (status == ESC || status == F10)
            break ;
        cur_fld = 1 ;
    }
    if (cur_fld == 2)
    {
        status = fld_sel (&gssem_var[0],&ssem_selv[0],ssem_
selv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            gens_buf.gens_sem = ssem_selv[0].cur_sv ;
        if (status == ESC || status == F10)
            break ;
        if (status==CUR_UP)
            cur_fld = 1 ;
        else
            cur_fld = 3 ;
    }
    if (cur_fld != 0)
        status = scr_mgr (OLD,ED_FLAG,OLD,gens_var,gens_lbl
,4,NULL,NULL) ;
    if (old_fld == 3)
    {
        st = gens_getsub() ;
        if (st == FALSE)
            cur_fld = 3 ;
    }
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    gens_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
    gens_print () ;
    gens_clr0 () ;
    return ;
}
instr ("การแจ้งเตือน....") ;
gens_key0.gens_dep = gens_buf.gens_dep ;
gens_key0.gens_sem = gens_buf.gens_sem ;
gens_key0.gens_year = gens_buf.gens_year ;
strcpy(gens_key0.code,gens_buf.code) ;
STAT1 = BTRV (B_GETEQ,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;

```

```

if (STAT1 != 0)
{
    gens_buf2rec ();
    STATUS = BTRV (B_INSERT,gens_blk,&gens_rec,&gens_len,&gens_
key0,0) ;
    if (STATUS != 0)
        b_error ("ไม่สามารถเก็บข้อมูลได้") ;
}
else
    b_error ("ไม่สามารถเก็บข้อมูลได้") ;
if (STATUS != 0 || STAT1 == 0)
{
    STATUS = BTRV (B_GETPR,gens_blk,&gens_rec,&gens_len,&gens_k
ey0,0) ;
    if (STATUS != 0)
        STATUS = BTRV (B_GETNX,gens_blk,&gens_rec,&gens_len
,&gens_key0,0) ;
    if (STATUS != 0)
        STATUS = BTRV (B_GETLW,gens_blk,&gens_rec,&gens_len
,&gens_key0,0) ;
    if (STATUS != 0)
    {
        clr_win (4,20,4,70) ;
        clr_win (5,41,5,50) ;
        scr_mgr (NEW,DISP_FLAG,NEW,gens_var,gens_lbl,4,NULL
,NULL) ;
        gens_clr0 () ;
        return ;
    }
}
gens_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
gens_print () ;
gens_clr0 () ;
}

gens_del ()
{
    int    sure ;

    sure = make_sure ("โปรดยืนยันการลบข้อมูล") ;
    if (sure==0)
        return ;
    instr ("กำลังลบข้อมูล....") ;
    STATUS = BTRV (B_DELETE,gens_blk,&gens_rec,&gens_len,&gens_key0,0)
;

    clrinstr () ;
    if (STATUS != 0)
    {
        scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
        gens_print () ;
        gens_clr0 () ;
        return ;
    }
    STATUS = BTRV (B_GETPR,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;

```

```
if (STATUS != 0)
    STATUS = BTRV (B_GETNX,gens_blk,&gens_rec,&gens_len,&gens_k
ey0,0) ;
    if (STATUS != 0)
    {
        clr_win (4,20,4,70) ;
        clr_win (5,41,5,50) ;
        scr_mgr (NEW,DISP_FLAG,NEW,gens_var,gens_lbl,4,NULL
,NULL) ;

        gens_clr0 () ;
        return ;
    }
    gens_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
    gens_print () ;
    gens_clr0 () ;
}
```



```

#include "extern.inc"
#include "stue.inc"

resume_stud ()
{
    int    func_no ;
    stud_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS == 0)
    {
        cpyd_stu () ;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stud_print () ;
        stud_datel () ;
        stud_date2 () ;
    }
    do
    {
        switch (func_no= stud_menu())
        {
            case F1 :
                stud_prev () ;
                break ;
            case F2 :
                stud_next () ;
                break ;
            case F3 :
                stud_srch () ;
                break ;
        }
    } while (func_no != F9) ;
}

stud_cpy ()
{
    strcpy (stu_rec.stu_id,stu_buf.stu_id) ;
    stu_rec.part = stu_buf.part;
    strcpy (stu_rec.th_name,stu_buf.th_name) ;
    stu_rec.sex = stu_buf.sex ;
    strcpy (stu_rec.th_sname,stu_buf.th_sname) ;
    strcpy (stu_rec.ename,stu_buf.ename) ;
    strcpy (stu_rec.e_sname,stu_buf.e_sname) ;
    strcpy (stu_rec.d_birth,stu_buf.d_birth) ;
    strcpy (stu_rec.d_cont,stu_buf.d_cont) ;
    strcpy (stu_rec.adv_name,stu_buf.adv_name);
    strcpy (stu_rec.add1,stu_buf.add1) ;
    strcpy (stu_rec.add2,stu_buf.add2) ;
    strcpy (stu_rec.tel,stu_buf.tel) ;
    strcpy (stu_rec.p_name,stu_buf.p_name) ;
    strcpy (stu_rec.p_sname,stu_buf.p_sname) ;
    strcpy (stu_rec.p_add1,stu_buf.p_add1) ;
    strcpy (stu_rec.p_add2,stu_buf.p_add2) ;
    strcpy (stu_rec.p_tel,stu_buf.p_tel) ;
}

```



```

}

stu_cpy1 ()
{
    stu_rec.stu_crad = 0 ;
    stu_rec.stu_sco = 0.0 ;
}

cpyd_stu ()
{
    strcpy (stu_buf.stu_id,stu_rec.stu_id) ;
    stu_buf.part = stu_rec.part;
    strcpy (stu_buf.th_name,stu_rec.th_name) ;
    stu_buf.sex = stu_rec.sex ;
    strcpy (stu_buf.th_sname,stu_rec.th_sname) ;
    strcpy (stu_buf.ename,stu_rec.ename) ;
    strcpy (stu_buf.e_sname,stu_rec.e_sname) ;
    strcpy (stu_buf.d_birth,stu_rec.d_birth) ;
    strcpy (stu_buf.d_cont,stu_rec.d_cont) ;
    strcpy (stu_buf.adv_name,stu_rec.adv_name);
    strcpy (stu_buf.add1,stu_rec.add1) ;
    strcpy (stu_buf.add2,stu_rec.add2) ;
    strcpy (stu_buf.tel,stu_rec.tel) ;
    strcpy (stu_buf.p_name,stu_rec.p_name) ;
    strcpy (stu_buf.p_sname,stu_rec.p_sname) ;
    strcpy (stu_buf.p_add1,stu_rec.p_add1) ;
    strcpy (stu_buf.p_add2,stu_rec.p_add2) ;
    strcpy (stu_buf.p_tel,stu_rec.p_tel) ;
}

stud_scr ()
{
    clrscr () ;
    frame ( 1, 2,21,80,"ประวัติส่วนตัวนักศึกษา") ;
    printp (9,39,"[dd/mm/yy]",NM_VDO) ;
    printp (10,39,"[dd/mm/yy]",NM_VDO) ;
    printp (19,39,"หมายเลขบัตร : ",NM_VDO) ;
}

stud_menu ()
{
    int i;
    static struct lbl_template fu_stud[] = {
        {23, 4,"1:ก.อน",RV_VDO},
        {23,20,"2:ผลฯ",RV_VDO},
        {23,35,"3:ค้นหา",RV_VDO},
        {23,53,"9:Exit",RV_VDO}
    };

    i = get_func (F3,fu_stud) ;
    return (i) ;
}

```

```

{
    char    buf[25] ;

    if (chk_date(stu_var[7].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (stu_var[7].datatag.str_add,buf) ;
    printp ( 9,53,"",NM_VDO) ;
    printp ( 9,53,buf,NM_HL_VDO) ;
}

stud_date2 ()
{
    char    buf[25];

    if (chk_date(stu_var[8].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (stu_var[8].datatag.str_add,buf) ;
    printp ( 10,53,"",NM_VDO) ;
    printp ( 10,53,buf,NM_HL_VDO) ;
}

stu_dep(cod,buf)
int cod;
char buff[];
{
    static char *dep [] = { "คณะวิศวกรรมศาสตร์และวิทยาการคอมพิวเตอร์","เคมี",
                             "ชีววิทยาประยุกต์", "ฟิสิกส์ประยุกต์",
                             "สังคมประยุกต์" };

    strcpy(buff,dep[cod]);
}

stud_print ()
{
    char    buf1[41] ,buf2[5] ,buf3[3] ;

    stu_dep (stu_buf.part,buf1);
    printp (stu_var[1].row,stu_var[1].col,buf1,NM_HL_VDO) ;
    printp (4,20,"",NM_HL_VDO) ;

    printp (4,20,buf1,NM_HL_VDO);
    stu_sex (stu_buf.sex,buf2) ;
    printp (stu_var[3].row,stu_var[3].col,buf2,NM_HL_VDO) ;
    printp (5,55,"",NM_HL_VDO);
    printp (5,55,buf2,NM_HL_VDO);
    printp (19,52,"",NM_HL_VDO);
    itoa_l (stu_rec.stu_crad,buf8,3) ;
    printp (19,52,buf3,NM_HL_VDO) ;
}

stu_sex (code,buf)
int code;
char    buf[] ;

```

```

{
static char *sex[] = {"ชาย",
                      "หญิง"} ;

strcpy (buf,sex[code]) ;
}

stud_prev ()
{
STATUS = BTRV (B_GETPR,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
if (STATUS == 0)
{
cpyd_stu () ;
scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
stud_print () ;
stud_date1 () ;
stud_date2 () ;
}
else
b_error ("กองบงกชหมดแล้วครับ") ;
return ;
}

stud_next ()
{
STATUS = BTRV (B_GETNX,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
if (STATUS == 0)
{
cpyd_stu () ;
scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
stud_print () ;
stud_date1 () ;
stud_date2 () ;
}
else
b_error ("กองบงกชหมดแล้วครับ") ;
return ;
}

stud_srch ()
{
int row,col,status ,new;
char buff[10];

T_E_flag = Eng ;
cur_fld = 0 ;
make_win (18, 6,22,36,"Search Window") ;
printw(20,9,"รหัสนักศึกษา",NM_HL_VDO);
strcpy(buff,stu_buf.stu_id);
row = stu_var[0].row;
col = stu_var[0].col;
stu_var[0].row = 20;
stu_var[0].col = 21;
new = NEW;
do {

```

```

        status = fld_edit(new,&stu_var[0],NAUTO_ADV,NULL,NULL) ;
        new = OLD;
    } while (status != RETURN && status != F10 && status !=ESC);
    stu_var[0].row = row ;
    stu_var[0].col = col;
    recall();
    if (status == ESC)
    {
        strcpy(stu_buf.stu_id,buff);
        return;
    }
    strcpy(stu_key0.id,stu_buf.stu_id);
    STATUS = BTRV(B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0);
    if (STATUS == 0)
    {
        cpyd_stu () ;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stud_print () ;
        stud_date1 () ;
        stud_date2 () ;
        return ;
    }
    else
        b_error ("ไม่พบรหัสนักศึกษาในแฟ้มของมขล") ;
    return ;
}

```

```

#include "extern.inc"
#include "stue.inc"
int    inpcrd[12] ;
input_grade ()
{
    int func_no ;
    inpf_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,inp_var,inp_lbl,16,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,inp_blk,&inp_rec,&inp_len,&inp_key0,0) ;
    if (STATUS==0)
    {
        inpf_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,inp_var,inp_lbl,16,NULL,NULL) ;
        inpf_print () ;
    }
    do {
        switch (func_no = inpf_menu())
        {
            case F1:
                inpf_oth () ;
                break ;
        }
    } while (func_no != F9) ;
}

inpf_scr ()
{
    clrscr () ;
    frame ( 2, 2,20,78,"ระบบประมวลผลผลการศึกษาระจาภาคการศึกษา") ;
    lineH (2,78,7) ;
    lineV (7,20,11) ;
    lineV (7,20,55) ;
    lineV (7,20,64) ;
    printp (4,13,"ชื่อ : ",NM_VDO) ;
    printp (7, 4,"รหัสวิชา",RV_VDO) ;
    printp (7,30,"ชื่อวิชา",RV_VDO) ;
    printp (7,57,"หน่วยกิต",RV_VDO) ;
    printp (7,67,"ผลการศึกษา",RV_VDO) ;
    printp (21,53,"รวม : ",NM_HL_VDO) ;
    printp (21,63,"เฉลี่ย : ",NM_HL_VDO) ;
}

inpf_buf2rec ()
{
    int    i ;

    strcpy (inp_rec.id,inp_buf.id) ;
    inp_rec.depp = inp_buf.depp ;
    inp_rec.sem = inp_buf.sem ;
    inp_rec.year = inp_buf.year ;
    for (i=0; i<12; i++)
    {

```

```

    }
    inp_rec.olddtot = inp_buf.olddtot ;
    inp_rec.oldsco = inp_buf.oldsco ;
    inp_rec.newtot = inp_buf.newtot ;
    inp_rec.newsco = inp_buf.newsco ;
    inp_rec.total = inp_buf.total ;
    inp_rec.sco = inp_buf.sco ;
}

inpf_rec2buf ()
{
    int    i ;

    strcpy (inp_buf.id,inp_rec.id) ;
    inp_buf.depp = inp_rec.depp;
    inp_buf.sem = inp_rec.sem ;
    inp_buf.year = inp_rec.year ;
    for (i=0; i<12; i++)
    {
        strcpy (inp_buf.sub[i].code,inp_rec.sub[i].code) ;
        strcpy (inp_buf.sub[i].grd,inp_rec.sub[i].grd) ;
    }
    inp_buf.olddtot = inp_rec.olddtot ;
    inp_buf.oldsco = inp_rec.oldsco ;
    inp_buf.newtot = inp_rec.newtot ;
    inp_buf.newsco = inp_rec.newsco ;
    inp_buf.total = inp_rec.total ;
    inp_buf.sco = inp_rec.sco ;
}

inpf_menu ()
{
    static struct lbl_template  fn_inp[] = { {23, 4,"1:ผลการศกษาของน.ภศ
๑กษา๑,๑",RV_VDO},
                                              {23,71,"9:Exit",RV_VDO}
    };

    int    i ;

    i = get_func (F1,fn_inp) ;
    return (i) ;
}

inpf_print ()
{
    inpf_name() ;
    inpf_dep();
    inpf_sem () ;
    inpf_year();
    inpf_getsub () ;
}

inpf_name ()
{

```

เอกสารนี้เป็นเอกสารที่ int ไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 char dbuf1[41],dbuf2[2] ;
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

strcpy (stu_key0.id,inp_buf.id) ;
STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่พบรหัสนักเรียนในแฟ้มข้อมูล") ;
    return (FALSE) ;
}
printw ( 4,18,"",NM_VDO)
;

printw ( 4,18,stu_rec.th_name,NM_HL_VDO) ;
i = 0 ;
while (stu_rec.th_name[i] != ' ')
    i++ ;
printw ( 4,19+i,stu_rec.th_sname,NM_HL_VDO) ;
return (TRUE) ;
}

depart(a,d)
int a;
char d[];
{
    static char *dep[] = {"คณะศึกษาศาสตร์และวิทยาการคอมพิวเตอร์",
        "เคมีอุตสาหกรรม",
        "เทคโนโลยีชีวภาพ",
        "ฟิสิกส์ประยุกต์",
        "สังคมประยุกต์"} ;
    strcpy(d,dep[a]);
}

inpf_dep ()
{
    char part2[41];

    depart(inp_buf.depp,part2);
    printw(inp_var[1].row,inp_var[1].col,part2,NM_HL_VDO);
    printw (5,18,"",NM_VDO);
    printw (5,18,part2,NM_HL_VDO);
    return (TRUE) ;
}

sema (ca,aa)
int ca;
char aa[];
{
    static char *sem[] = {"1st",
        "2nd",
        "Summer" } ;
    strcpy(aa,sem[ca]);
}

inpf_sema ()
{
    char sbuf[7];

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    sema(inp_buf.sem,sbuf);
    printp(inp_var[2].row,inp_var[2].col,sbuf,NM_HL_VDO);
    printp(6,18,"      ",NM_VDO);
    printp(6,18,sbuf,NM_HL_VDO) ;
    return(TRUE);
}

inpf_year ()
{
    char    ybuf[2] ;

    printp ( 6,34," ",NM_VDO) ;
    itoa_l (inp_buf.year,ybuf,1) ;
    printp ( 6,34,ybuf,NM_HL_VDO) ;
}

inpf_getsub ()
{
    char    gbuf[2] ,ybuf[2] ;
    char    gbuf1[61] ;
    char    tbuf[3] ;
    int     crd, n, i ;

    crd = 0 ;
    for (i=0; i<12; i++)
    {
        if (inp_buf.sub[i].code[0] != ' ')
        {
            strcpy (osub_key0.osub_id,inp_buf.sub[i].code) ;
            STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len
,&osub_key0,0) ;

            strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
            STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len
,&subj_key0,0) ;

            printp (8+i,4,subj_rec.subj_code,NM_HL_VDO) ;
            strcpy (gbuf1,subj_rec.subj_name) ;
            n = 0 ;
            while (gbuf1[n] != ' ' || gbuf1[n+1] != ' ')
                n++ ;
            gbuf1[n] = '\0' ;
            printp (8+i,13,"

",NM_VDO) ;

            printp (8+i,13,gbuf1,NM_HL_VDO) ;
            itoa_l (subj_rec.subj_cred,gbuf,1) ;
            printp (8+i,60,gbuf,NM_HL_VDO) ;
            inpcrd[i] = subj_rec.subj_cred ;
            crd += subj_rec.subj_cred ;
        }
    }

    printp (21,59," ",NM_VDO) ;
    itoa_l (crd,tbuf,2) ;
    printp (21,59,tbuf,NM_HL_VDO) ;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[illegible]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ **status = ESC ;** ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
iflag = 1 ;
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    else {
        strcpy (mon_key0.id,inp_buf.id) ;
        mon_key0.depp = inp_buf.depp;
        mon_key0.sem = inp_buf.sem ;
        mon_key0.year = inp_buf.year ;
        STATUS = BTRV (B_GETEQ,mon_blk,&mon_rec,&mo
n_len,&mon_key0,0) ;

        if (STATUS != 0)
        {
            b_error( "ไม่พบรหัสสนุภศกษานในแฟ้ม
มข๖๐๗๖๗" ) ;

            cur_fld = 0 ;
            status = RETURN ;
        }
        else {
            inpf_mon2inp () ;
            inpf_print();
            status = F10 ;
            iflag = 2 ;
        }
    }
} while (status != ESC && status != F10) ;
if (status == ESC)
    return ;
cnts = 3 ;
for (i=0; i<12; i++)
{
    if (inp_buf.sub[i].code[0] != ' ')
        cnts++ ;
}
cur_fld = 4 ;
sfld = cur_fld ;
printp (4+sfld,4,inp_buf.sub[sfld-4].code,RV_HL_VDO) ;
do {
    if (cur_fld < 4)
        cur_fld = cnts ;
    if (cur_fld > cnts)
        cur_fld = 4 ;
    if (sfld != cur_fld)
    {
        if (inpf_chk(sfld-4)==TRUE)
        {
            inpf_avg () ;
            printp (4+sfld,4,inp_buf.sub[sfld-4].code,N
M_HL_VDO) ;

            sfld = cur_fld ;
            printp (4+sfld,4,inp_buf.sub[sfld-4].code,R
V_HL_VDO) ;
        }
    }
} else {
    b_error ("ไม่พบเกรตม๑๑") ;
    cur_fld = sfld ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    status = scr_mgr (OLD,ED_FLAG,OLD,inp_var,inp_lbl,16,NULL,N
ULL) ;

    if (status == F10)
    {
        cnts -= 3 ;
        for (i=0; i<cnts; i++)
        {
            if (inp_buf.sub[i].grd[0] == ' ')
            {
                error ("ยังป้อนผลการศกษาไม่ครบ") ;
                cur_fld = i+1 ;
                sfld = i+1;
                status = RETURN ;
                break ;
            }
        }
    }
} while (status != F10 && status != ESC) ;
if (status == ESC)
    return ;
inpf_newavg () ;
inpf_buf2rec () ;
if (iflag != 1)
    STATUS = BTRV (B_INSERT,inp_blk,&inp_rec,&inp_len,&inp_key0
,0) ;
    if (STATUS != 0)
        b_error ("ไม่สามารถแก้ไขข้อมูลได้") ;
return ;
}

inpf_fld (stat)
int *stat ;
{
    if (*stat == ESC)
        return ;
    if (old_fld == 0 || *stat == F10)
    {
        if ((inpf_name()) == FALSE)
        {
            *stat = RETURN ;
            cur_fld = 0 ;
            return ;
        }
    }
    if (old_fld == 1 || *stat == F10 )
    {
        if (inp_buf.depp > 4 )
        {
            error ("โปรดระบ ภาควชา");
            cur_fld = old_fld;
            *stat = RETURN ;
            return;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (old_fld ==2 !! *stat == F10)
{
    if (inp_buf.sem > 2)
    {
        error("โปรแกรม ภาควิชาการศึกษา");
        cur_fld = old_fld ;
        *stat = RETURN ;
        return ;
    }
}

}

inpf_mon2inp ()
{
    int    i ;

    strcpy(inp_buf.id,mon_rec.id);
    inp_buf.depp = mon_rec.depp;
    inp_buf.sem = mon_rec.sem ;
    inp_buf.year = mon_rec.year ;
    for (i=0; i<12; i++)
        strcpy(inp_buf.sub[i].code,mon_rec.sub[i].code) ;
}

inpf_newavg ()
{
    float  avg, score = 0.0 ;
    int    i, sum = 0 ;
    char    abuf1[5], abuf2[3], abuf3[3], abuf4[5] ;

    for (i=0; i<12; i++)
    {
        if (inp_buf.sub[i].grd[0] != ' ')
        {
            sum += inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"A ")==0)
                score += 4.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"B+")==0)
                score += 3.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"B ")==0)
                score += 3.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"C+")==0)
                score += 2.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"C ")==0)
                score += 2.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"D+")==0)
                score += 1.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"D ")==0)
                score += 1.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"F ")==0)
                score += 0.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"I ")==0)
            {
                score += 0.0 * (float)inpcrd[i] ;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

        sum -= inpcrd[i] ;
    }
}

inp_buf.newtot += sum ;
inp_buf.newsco += score ;
avg = score/(float)sum ;
ftoa_1 (avg,abuf1,(float)4.22) ;
printp (21,70,abuf1,NM_HL_VDO) ;
strcpy (stu_key0.id,inp_buf.id) ;
STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
printp (21,5," ",NM_VDO) ;
ftoa_1 (inp_buf.newsco,abuf2,(float)4.22) ;
printp (21,5,abuf2,NM_HL_VDO) ;
printp (21,15," ",NM_VDO) ;
itoa_1 (inp_buf.newtot,abuf3,2) ;
printp (21,15,abuf3,NM_HL_VDO) ;
inp_buf.oldtot += stu_rec.stu_crad ;
inp_buf.oldsco += stu_rec.stu_sco ;
stu_rec.stu_crad += sum ;
stu_rec.stu_sco += score ;
inp_buf.total += stu_rec.stu_crad ;
inp_buf.sco += stu_rec.stu_sco ;
printp (21,20," ",NM_VDO) ;
atoa_1 (inp_buf.sco,abuf4,(float)4.22) ;
printp (21,20,abuf4,NM_HL_VDO) ;
STATUS = BTRV (B_UPDATE,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
}

inp_fchk (int sc)
{
    if (strcmp (inp_buf.sub[sc].grd,"A ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"B+")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"B ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"C+")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"C ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"D+")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"D ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"F ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"I ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"S ")==0)
        return (TRUE) ;
    if (strcmp (inp_buf.sub[sc].grd,"U ")==0)
        return (TRUE) ;
    return (FALSE) ;
}

```

```
inpf_avg ()
{
    float  avg, score = 0.0 ;
    int    i , sum = 0 ;
    char   abuf[5] ;

    for (i=0; i<12; i++)
    {
        if (inp_buf.sub[i].grd[0] != ' ')
        {
            sum += inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"A ")==0)
                score += 4.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"B+")==0)
                score += 3.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"B ")==0)
                score += 3.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"C+")==0)
                score += 2.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"C ")==0)
                score += 2.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"D+")==0)
                score += 1.5 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"D ")==0)
                score += 1.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"F ")==0)
                score += 0.0 * (float)inpcrd[i] ;
            if (strcmp (inp_buf.sub[i].grd,"I ")==0)
            {
                score += 0.0 * (float)inpcrd[i] ;
                sum -= inpcrd[i] ;
            }
        }
    }
    avg = score/(float)sum ;
    ftoa_1 (avg,abuf,(float)4.22) ;
    printp (21,70,abuf,NM_HL_VDO) ;
}
```

```

#include "extern.inc"
#include "stue.inc"

conclude_semester ()
{
    int      func_no ;

    csemf_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,csem_var,csem_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,csem_blk,&csem_rec,&csem_len,&csem_key0,0) ;
    if (STATUS==0)
    {
        csemf_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,csem_var,csem_lbl,4,NULL,NULL) ;
        csemf_print () ;
    }
    do {
        switch (func_no = csemf_menu())
        {
            case F1:
                csemf_oth () ;
                break ;
        }
    } while (func_no != F9) ;
}

csemf_scr ()
{
    clrscr () ;
    frame (6,4,16,79,"สรุปผลการศึกษาระดับภาคเรียน") ;
    printp (8,11,"ชื่อ-สกุล : ",NM_VDO) ;
    lineH (5,78,11) ;
    lineV (11,16,28) ;
    lineV (11,16,45) ;
    lineV (11,16,62) ;
    printp (11,32,"เฉพาะเทอมนี้",RV_VDO) ;
    printp (11,51,"ของเดิม",RV_VDO) ;
    printp (11,68,"ของใหม่",RV_VDO) ;
    printp (13,9,"จำนวนหน่วยกิต",NM_VDO) ;
    printp (14,9,"เกรดเฉลี่ย (GPS/GPA)",NM_VDO) ;
    printp (17,11,"STATUS :",NM_VDO) ;
}

csemf_menu ()
{
    static struct lbl_template  fn_csem[] = { {23, 4,"1:สรุปผลการศึกษา",RV_
VDO},
                                                {23,71,"9:Exit",RV_VDO}
    } ;

    int      i ;

    i = get_func (F1,fn_csem) ;
    return (i) ;
}

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ภายใต้การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

csemf_inp2csem ()
{
    strcpy (csem_buf.code,inp_rec.id) ;
    csem_buf.dep = inp_rec.depp ;
    csem_buf.sem = inp_rec.sem ;
    csem_buf.year = inp_rec.year ;
}

csemf_buf2rec ()
{
    strcpy (csem_rec.code,csem_buf.code) ;
    csem_rec.dep = csem_buf.dep;
    csem_rec.sem = csem_buf.sem ;
    csem_rec.year = csem_buf.year ;
}

csemf_rec2buf ()
{
    strcpy (csem_buf.code,csem_rec.code) ;
    csem_buf.dep = csem_rec.dep;
    csem_buf.sem = csem_rec.sem ;
    csem_buf.year = csem_rec.year ;
    csem_buf.ocra = csem_rec.ocra ;
    csem_buf.scra = csem_rec.scra ;
    csem_buf.cra = csem_rec.cra ;
    csem_buf.gps = csem_rec.gps ;
    csem_buf.gpa = csem_rec.gpa ;
    csem_buf.ngpa = csem_rec.ngpa ;
}

csemf_print ()
{
    csemf_name() ;
    csemf_dep() ;
    csemf_sem () ;
    csemf_year() ;
    csemf_getsum () ;
}

csemf_name ()
{
    int    depp ,i ;
    char   dbuf1[41],dbuf2[2] ;

    strcpy (stu_key0.id,csem_buf.code) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบรหัสส่วนบุคคลนักเรียนแฟ้มขอมูล") ;
        return (FALSE) ;
    }
    if (stu_rec.sex == 0)
        printf(8,22,"นาย",NM_HL_VDO) ;
    else

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        printp(8,22,"น.ส.",NM_HL_VDO) ;
        printp (8,26,"                                ",NM_VDO)
    ) ;
    printp (8,26,stu_rec.th_name,NM_HL_VDO) ;
    i = 0 ;
    while (stu_rec.th_name[i] != ' ')
        i++ ;
    printp (8,27+i,stu_rec.th_sname,NM_HL_VDO) ;
    return (TRUE) ;
}

depart1 (a,d)
int    a;
char   d[];
{
    static char *dep[] = {"คณะศึกษาศาสตร์และวิทยาการคอมพิวเตอร์",
                           "คณะอุตสาหกรรม",
                           "เทคโนโลยีชีวภาพ",
                           "ฟิสิกส์ประยุกต์",
                           "สถิติประยุกต์"} ;

    strcpy(d,dep[a]);
}

csemf_dep ()
{
    char    part2[41];

    depart1 (csem_buf.dep,part2);
    printp(csem_var[1].row,csem_var[1].col,part2,NM_HL_VDO);
    printp (9,22,"                                ",NM_VDO);
    printp (9,22,part2,NM_HL_VDO);
    return (TRUE) ;
}

sema1 (ca,aa)
int    ca;
char   aa[];
{
    static char    *sem[] = {"1st",
                              "2nd",
                              "Summer" } ;

    strcpy(aa,sem[ca]);
}

csemf_sem ()
{
    char    sbuf[7] ;

    sema1 (csem_buf.sem,sbuf);
    printp(csem_var[2].row,csem_var[2].col,sbuf,NM_HL_VDO);
    printp(10,22,"                                ",NM_VDO);
    printp(10,22,sbuf,NM_HL_VDO) ;
    return(TRUE);
}

```

```

}
```

```

csemf_year ()
{
    char    ybuf[2] ;

    printp (10,45," ",NM_VDO) ;
    itoa_l (csem_buf.year,ybuf,1) ;
    printp (10,45,ybuf,NM_HL_VDO) ;
}

```

```

csemf_oth ()
{
    int    i , status, sure, stat1 ;

    scr_mgr (NEW,DISP_FLAG,NEW,csem_var,csem_lbl,4,NULL,NULL) ;
    clr_win (7,22,9,76) ;
    clr_win (10,22,10,35) ; clr_win (10,45,10,50) ;
    clr_win (13,32,14,40) ; clr_win (13,49,14,55) ;
    clr_win (13,66,14,72) ; clr_win (18,22,18,40) ;
    T_E_flag = Eng ;
    cur_fld = 0 ;
    do{
        if (cur_fld == 1)
        {
            T_E_flag = Eng;
            status = fld_sel (&csemdep_var[0],&dep_selv[0],dep_
selv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                csem_buf.dep = dep_selv[0].cur_sv ;
            cur_fld = 2 ;
        }
        if (cur_fld==2)
        {
            T_E_flag = Eng;
            status = fld_sel (&csemsem_var[0],&ipsem_selv[0],ip
sem_selv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                csem_buf.sem = ipsem_selv[0].cur_sv ;
            if (status==CUR_UP)
                cur_fld = 0 ;
            else
                cur_fld = 3 ;
        }
        status = scr_mgr (OLD,ED_FLAG,OLD,csem_var,csem_lbl,4,NULL,
NULL) ;
        csemf_fld(&status) ;
    } while (status != ESC && status != F10) ;
    if (status == ESC)
    {
        csemf_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,csem_var,csem_lbl,4,NULL,NULL) ;
        csemf_print () ;
        return ;
    }
}

```



```

    }
    strcpy (csem_key0.code,csem_buf.code) ;
    csem_key0.dep = csem_buf.dep;
    csem_key0.sem = csem_buf.sem ;
    csem_key0.year = csem_buf.year ;
    stat1 = BTRV (B_GETEQ,csem_blk,&csem_rec,&csem_len,&csem_key0,0) ;
    if (stat1 != 0)
    {
        sure = make_sure ("โปรดขออนุญาตการแก้ไขข้อมูล") ;
        if (sure == 0)
            return ;
        instr ("การแก้ไขข้อมูล....") ;
        csemf_buf2rec () ;
        STATUS = BTRV (B_INSERT,csem_blk,&csem_rec,&csem_len,&csem_
key0,0) ;
        if (STATUS != 0)
            b_error ("ไม่สามารถแก้ไขข้อมูลเข้าแฟ้มได้") ;
    }
    else
        b_error ("ไม่สามารถแก้ไขข้อมูลเข้าแฟ้มได้") ;
    if (stat1 != 0 || STATUS != 0)
    {
        STATUS = BTRV (B_GETPR,csem_blk,&csem_rec,&csem_len,&csem_k
ey0,0) ;
        if (STATUS != 0)
            STATUS = BTRV (B_GETNX,csem_blk,&csem_rec,&csem_len,&csem_k
ey0,0) ;
        if (STATUS != 0)
        {
            scr_mgr (NEW,DISP_FLAG,NEW,csem_var,csem_lbl,4,NULL
,NULL) ;
            return ;
        }
    }
    csemf_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,csem_var,csem_lbl,4,NULL,NULL) ;
    csemf_print () ;
}

```

```

csemf_fld (stat)
int *stat ;
{
    if (*stat == ESC)
        return ;
    if (old_fld == 0 || *stat == F10)
    {
        if ((csemf_name()) == FALSE)
        {
            *stat = RETURN ;
            cur_fld = 0 ;
            return ;
        }
    }
    if (old_fld == 1 || *stat == F10 )

```

```

        if (csem_buf.dep > 4 )
        {
            error("โปรแกรมผิดพลาด");
            cur_fld = old_fld;
            *stat = RETURN ;
            return;
        }
    }
    if (old_fld == 2 || *stat == F10)
    {
        if (csem_buf.sem > 2)
        {
            error("โปรแกรมผิดพลาด");
            cur_fld = old_fld ;
            *stat = RETURN ;
            return ;
        }
    }
    if (old_fld == 3 || *stat == F10)
    {
        if ((csemf_getsum ()) == FALSE)
        {
            *stat = RETURN ;
            cur_fld = 0 ;
            return ;
        }
    }
}

csemf_getsum ()
{
    char    grbuf1[3], grbuf2[3], grbuf3[3] ;
    char    abuf1[5], abuf2[5], abuf3[5] ;

    clr_win (18,22,18,40) ;
    strcpy (inp_key0.id,csem_buf.code) ;
    inp_key0.depp = csem_buf.dep;
    inp_key0.sem = csem_buf.sem ;
    inp_key0.year = csem_buf.year ;
    STATUS = BTRV (B_GETEQ,inp_blk,&inp_rec,&inp_len,&inp_key0,0) ;
    if (STATUS != 0)
    {
        b_error ( "ไม่พบข้อมูลงานแฟ้ม" ) ;
        return (FALSE) ;
    }
    csem_buf.scra = inp_rec.newtot ;
    printp (13,37,"      ",NM_VDO) ;
    itoa_1 (csem_buf.scra,grbuf1,2) ;
    printp (13,37,grbuf1,NM_HL_VDO) ;

    csem_buf.ocra = inp_rec.oldtot ;
    printp (13,54,"      ",NM_VDO) ;
    itoa_1 (csem_buf.ocra,grbuf2,2) ;
    printp (13,54,grbuf2,NM_HL_VDO) ;
}

```

```

csem_buf.cra = inp_rec.total ;
printp (13,71,"      ",NM_VDO) ;
itoa_1 (csem_buf.cra,grbuf3,2) ;
printp (13,71,grbuf3,NM_HL_VDO) ;

if (inp_rec.newtot < 1)
{
    b_error ("หน่วยกิตรวมค.ไม่เป็น 0") ;
    return (FALSE) ;
}
csem_buf.gps = inp_rec.newsco / (float)inp_rec.newtot ;
printp (14,35,"      ",NM_VDO) ;
itoa_1 (csem_buf.gps,abuf1,(float)4.22) ;
printp (14,35,abuf1,NM_HL_VDO) ;

if (inp_rec.oldtota < 1)
{
    b_error ("หน่วยกิตรวมค.ไม่เป็น 0") ;
    return (FALSE) ;
}
csem_buf.gpa = inp_rec.oldsco / (float)inp_rec.oldtota ;
printp (14,52,"      ",NM_VDO) ;
ftoa_1 (csem_buf.gpa,abuf2,(float)4.22) ;
printp (14,52,abuf2,NM_HL_VDO) ;

if (inp_rec.total < 1)
{
    b_error ("หน่วยกิตรวมค.ไม่เป็น 0") ;
    return (FALSE) ;
}
csem_buf.ngpa = inp_rec.sco / (float)inp_rec.total ;
printp (14,69,"      ",NM_VDO) ;
ftoa_1 (csem_buf.ngpa,abuf3,(float)4.22) ;
printp (14,69,abuf3,NM_HL_VDO) ;

if (csem_buf.ngpa < 2.0)
    printp (187,22,"PROBATION",NM_HL_VDO) ;
else
    printp (17,22,"PASS",NM_HL_VDO) ;

csem_rec.ocra = csem_buf.ocra ;
csem_rec.scra = csem_buf.scra ;
csem_rec.cra = csem_buf.cra ;
csem_rec.gps = csem_buf.gps ;
csem_rec.gpa = csem_buf.gpa ;
csem_rec.ngpa = csem_buf.ngpa ;
return (TRUE) ;
}

```

```

#include "extern.inc"
#include "stue.inc"

te_egrđ ()
{
    int      func_no ;

    tgr_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,tgr_var,tgr_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS == 0)
    {
        tgr_buff () ;
        scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
        tgr_getsub () ;
        tgr_name () ;
    }
    do
    {
        switch (func_no = tgr_menu())
        {
            case F1:
                tgr_prev () ;
                break ;
            case F2:
                tgr_next () ;
                break ;
            case F3:
                tgr_srch () ;
                break ;
            case F4:
                tgr_edit () ;
                break ;
            case F5:
                tgr_ins () ;
                break ;
            case F6:
                tgr_del () ;
                break ;
        }
    } while (func_no != F9) ;
}

tgr_scr ()
{
    clrscr() ;
    frame (3,5,15,75,"การบันทึกการเกิดเฉพาะรายวิชา") ;
    printp (5,17,"ชื่อวิชา :",NM_VDO) ;
    printp (6,12,"อาจารย์ผะสอน :",NM_VDO) ;
    lineH (5,75,7) ;
    printp (9,20,"ชื่อ :",NM_VDO) ;
    printp (10,16,"ภาควชา :",NM_VDO);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    strcpy (tgr_buf.tgr_sub,tgr_rec.tgr_sub) ;
    strcpy (tgr_buf.code,tgr_rec.code) ;
    strcpy (tgr_buf.gradel,tgr_rec.gradel) ;
    strcpy (tgr_buf.grade2,tgr_rec.grade2) ;
}

tgr_recc ()
{
    strcpy (tgr_rec.tgr_sub,tgr_buf.tgr_sub) ;
    strcpy (tgr_rec.code,tgr_buf.code) ;
    strcpy (tgr_rec.gradel,tgr_buf.gradel) ;
    strcpy (tgr_rec.grade2,tgr_buf.grade2) ;
}

tgr_name ()
{
    int    i;
    char   part2[41];

    strcpy (stu_key0.id,tgr_buf.code) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS!=0)
    {
        bt_error () ;
        return (FALSE) ;
    }
    printp ( 9,25,"                                ",NM_VDO)
;
    printp ( 9,25,stu_rec.th_name,NM_HL_VDO) ;
    i = 0 ;
    while (stu_rec.th_name[i] != ' ')
        i++ ;
    printp ( 9,27+i,stu_rec.th_sname,NM_HL_VDO) ;

    stu_dep(stu_buf.part,part2);
    printp (10,25,"                                ",NM_VDO);
    printp (10,25,part2,NM_HL_VDO);
    return (TRUE) ;
}

tgr_getsub ()
{
    strcpy (osub_key0.osub_id,tgr_buf.tgr_sub) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS!=0)
    {
        bt_error();
        b_error ("ไม่พบการเปิดสอนวิชา") ;
        return (FALSE) ;
    }
    printp (6,25,"                                ",NM_VDO);
    printp (6,25,osub_rec.osub_tea,NM_HL_VDO);
    strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);

```

```

    if (STATUS != 0)
        return (FALSE) ;
    printp (5,25,"",NM_VDO);
    printp (5,25,subj_rec.subj_name,NM_HL_VDO) ;
    return (TRUE) ;
}

tgr_menu ()
{
    static struct lbl_template fn_tgr[] = {
        {23, 4,"1:ก.อน",RV_VDO},
        {23,11,"2:ทล.",RV_VDO},
        {23,18,"3:คณหา",RV_VDO},
        {23,26,"4:แภ.",RV_VDO},
        {23,35,"5:เพ.ม.ต.",RV_VDO},
        {23,46,"6:ลบ",RV_VDO},
        {23,71,"9:Exit",RV_VDO}
    } ;

    int i ;

    i = get_func (F6,fn_tgr) ;
    return (i) ;
}

tgr_prev ()
{
    STATUS = BTRV (B_GETPR,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ก.ง.บ.ม.จ.ร.ก.แภ.ล.ว.ค.ร.บ") ;
        return ;
    }
    tgr_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub () ;
    tgr_name () ;
}

tgr_next ()
{
    STATUS = BTRV (B_GETNX,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ก.ง.บ.ม.จ.ร.ส.ค.ท.ว.ย.ล.ว.ค.ร.บ") ;
        return ;
    }
    tgr_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub () ;
    tgr_name () ;
}

tgr_srch ()
{
    int status, new, row, col ;
    char buff[7] ;

```



```

T_E_flag = Eng ;
make_win (18,6,22,40,"Search Window") ;
printw (20,16,"รหัสผ่าน : ",NM_HL_VDO) ;
strcpy (buff,tgr_buf.tgr_sub) ;
row = tgr_var[0].row ;
col = tgr_var[0].col ;
tgr_var[0].row = 20 ;
tgr_var[0].col = 24 ;
new = NEW ;
do
{
    status = fld_edit (new,&tgr_var[0],NAUTO_ADV,NULL,NULL) ;
    new = OLD ;
} while (status != RETURN && status != F10 && status != ESC) ;
tgr_var[0].row = row ;
tgr_var[0].col = col ;
recall () ;
if (status == ESC)
{
    strcpy (tgr_buf.tgr_sub,buff) ;
    return ;
}
strcpy (tgr_key0.tgr_sub,tgr_buf.tgr_sub) ;
STATUS = BTRV (B_GETEQ,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่พบรหัสผ่านแฟ้มขอมูล") ;
    return ;
}
tgr_buff () ;
scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
tgr_getsub () ;
tgr_name () ;
}

tgr_edit ()
{
    int      status ;

    cur_fld = 0 ;
    T_E_flag = Eng ;
    STATUS=BTRV(B_GETEQL,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0);
    do
    {
        switch (cur_fld)
        {
            default :
                T_E_flag = Eng ;
                break;
        }
        status = scr_mgr (OLD,ED_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NU
LL) ;
        tgr_fld (&status) ;
    } while (status != F10 && status != ESC) ;
    if (status == ESC)

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของสำนักงานคณะกรรมการ
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    STATUS=BTRV(B_UNLOCK,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0)
;
    tgr_buff ();
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub ();
    tgr_name ();
    return ;
}
tgr_recc ();
instr (0,"การแก้ไขข้อมูล....") ;
STATUS = BTRV (B_UPDATE,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถแก้ไขข้อมูลได้") ;
    STATUS = BTRV (B_GETGE,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,
0) ;
    if (STATUS != 0)
        scr_mgr (NEW,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,N
ULL) ;
    else
    {
        tgr_buff ();
        scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,N
ULL) ;
        tgr_getsub ();
        tgr_name ();
    }
    return ;
}
STATUS=BTRV(B_UNLOCK,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0);
scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
tgr_getsub ();
tgr_name ();
}

tgr_ins ()
{
    int      status ;

    printp (5,25,"                  ",NM_VDO);
    printp (6,25,"                  ",NM_VDO) ;
    printp (9,25,"                  ",NM_VDO) ;
    printp (10,25,"                  ",NM_VDO);
    scr_mgr (NEW,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    cur_fld = 0 ;
    T_E_flag = Eng ;

    do
    {
        switch (cur_fld)
        {
            default :
                T_E_flag = Eng ;
                break;
        }
    }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    status = scr_mgr (OLD,ED_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NU
LL) ;
    tgr_fld (&status) ;
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    tgr_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub () ;
    tgr_name () ;
    return ;
}
tgr_recc () ;
instr (0,"การลบข้อมูล...") ;
STATUS = BTRV (B_INSERT,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถลบข้อมูล...") ;
    STATUS = BTRV (B_GETPR,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,
0) ;
    if (STATUS!=0)
        STATUS = BTRV (B_GETNX,tgr_blk,&tgr_rec,&tgr_len,&t
gr_key0,0) ;
    if (STATUS!=0)
        scr_mgr (NEW,DISP_FLAG,NEW,tgr_var,tgr_lbl,4,NULL,N
ULL) ;
}
scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
tgr_getsub () ;
tgr_name () ;
}

tgr_del ()

{
    int    sure ;

    sure = make_sure ("โปรดหยุดการลบข้อมูล") ;
    if (sure == 0)
        return ;
    STATUS=BTRV(B_GETEQL,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0);
    instr ( 0,"การลบข้อมูล..." ) ;
    STATUS = BTRV ( B_DELETE,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0 ) ;
    clrinstr () ;
    if ( STATUS != 0 )
    {
        scr_mgr ( OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL ) ;
        tgr_getsub () ;
        tgr_name () ;
        return ;
    }
    STATUS=BTRV(B_UNLOCK,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0);
    STATUS = BTRV ( B_GETNX,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0 ) ;
    if (STATUS != 0)

```

```

STATUS = BTRV( B_GETPR,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
if (STATUS != 0)
{
    scr_mgr (NEW,DISP_FLAG,NEW,tgr_var,tgr_lbl,4,NULL,NULL) ;
    return ;
}
tgr_buff () ;
scr_mgr (NEW,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
}

tgr_fld (st)
int *st ;
{
    if (*st==ESC)
        return ;
    if (old_fld==0 && cur_fld==1)
    {
        if (tgr_getsub () == FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
    }
    if (old_fld==1 && cur_fld==2)
    {
        if (tgr_name () == FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
    }
    return ;
}
}

```

```

#include "extern.inc"
#include "stue.inc"

re_edu ()
{
    int      func_no ;

    resd_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,res_var,res_lbl,9,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,res_blk,&res_rec,&res_len,&res_key0,0) ;
    if (STATUS==0)
    {
        resd_buff () ;
        scr_mgr (OLD,DISP_FLAG,OLD,res_var,res_lbl,9,NULL,NULL) ;
        resd_name () ;
    }
    do {
        switch (func_no = resd_menu())
        {
            case F1:
                resd_prev () ;
                break ;
            case F2:
                resd_next () ;
                break ;
            case F3:
                resd_srch () ;
                break ;
        }
    } while (func_no!=F9) ;
}

resd_scr ()
{
    clrscr();
    frame (3,2,14,79,"ขอมองลประวัติการศึกษา") ;
    printp(5,11,"ชื่อ : ",NM_VDO) ;
    printp(6,7,"ภาควชา : ",NM_VDO) ;
    lineH (2,79,7) ;
    lineV (7,14,22) ;
    lineV (7,14,67) ;
    printp(7,6,"ระดับการศึกษา",RV_VDO);
    printp(7,40,"สถานศึกษา",RV_VDO);
    printp(7,68,"สิ่งแวดล้อมการศึกษา",RV_VDO);
    printp(9,4,"ประเภทวิชาตอนต้น",NM_VDO);
    printp(10,4,"ประเภทวิชาตอนปลาย",NM_VDO);
    printp(11,4,"มัธยมศึกษาตอนต้น",NM_VDO);
    printp(12,4,"มัธยมศึกษาตอนปลาย",NM_VDO);
}

resd_buff ()
{
    strcpy (res_buf.stu_id,res_rec.stu_id) ;
    strcpy (res_buf.locate1,res_rec.locate1) ;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการใช้งานเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

strcpy (res_buf.time_int1,res_rec.time_int1) ;
strcpy (res_buf.locate2,res_rec.locate2);
strcpy (res_buf.time_int2,res_rec.time_int2);
strcpy (res_buf.locate3,res_rec.locate3) ;
strcpy (res_buf.time_int3,res_rec.time_int3) ;
strcpy (res_buf.locate4,res_rec.locate4);
strcpy (res_buf.time_int4,res_rec.time_int4);
}

resd_recc ()
{
    strcpy (res_rec.stu_id,res_buf.stu_id) ;
    strcpy (res_rec.locate1,res_buf.locate1) ;
    strcpy (res_rec.time_int1,res_buf.time_int1) ;
    strcpy (res_rec.locate2,res_buf.locate2);
    strcpy (res_rec.time_int2,res_buf.time_int2);
    strcpy (res_rec.locate3,res_buf.locate3) ;
    strcpy (res_rec.time_int3,res_buf.time_int3) ;
    strcpy (res_rec.locate4,res_buf.locate4);
    strcpy (res_rec.time_int4,res_buf.time_int4);
}

resd_name ()
{
    char    name[21];
    char    sname[31];
    char    part2[41];
    int     ncnt = 0 ;

    strcpy (stu_key0.id,res_buf.stu_id) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS != 0)
    {
        bt_error ("ไม่พบรหัสบันทึกศึกษานี้ในแฟ้มข้อมูล") ;
        return (FALSE) ;
    }
    if (stu_rec.sex == 0)
        printp (5,16,"นาย",NM_VDO) ;
    else
        printp (5,16,"น.ส.",NM_VDO) ;
    strcpy (name,stu_rec.th_name) ;
    while (name[ncnt] != ' ' ; name[ncnt] != ' ')
        ncnt++ ;
    name[ncnt] = '\0' ;
    printp (5,20,"",NM_VDO) ;
    printp (5,20,name,NM_VDO) ;
    strcpy (sname,stu_rec.th_sname);
    sname[25] = '\0' ;
    printp (5,21+ncnt,"",NM_VDO) ;
    printp (5,21+ncnt,sname,NM_VDO) ;
    part2[39] = '\0';
    stu_dep (stu_buf.part,part2);
    printp (6,16,"",NM_VDO);
    printp (6,16,part2,NM_VDO);
    return (TRUE);
}

```



```

}

resd_menu ()
{
static struct lbl_template fnd_res[] = {
                                {23, 4,"1:กลับ",RV_VDO},
                                {23,20,"2:ลบ",RV_VDO},
                                {23,35,"3:ค้นหา",RV_VDO},
                                {23,53,"9:Exit",RV_VDO}
                                } ;

    int    i ;

    i = get_func (F3,fnd_res) ;
    return (i) ;
}

resd_prev ()
{
    STATUS = BTRV (B_GETPR,res_blk,&res_rec,&res_len,&res_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("กดปุ่มย้อนกลับแล้วครับ") ;
        return ;
    }
    resd_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,res_var,res_lbl,9,NULL,NULL) ;
    resd_name () ;
}

resd_next ()
{
    STATUS = BTRV (B_GETNX,res_blk,&res_rec,&res_len,&res_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("กดปุ่มขอยกสื่อก่อนแล้วครับ") ;
        return ;
    }
    resd_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,res_var,res_lbl,9,NULL,NULL) ;
    resd_name () ;
}

resd_srch ()
{
    int    status, new, row, col ;
    char    buff[10] ;

    T_E_flag = Eng ;
    make_win (18, 6,22,36,"Search Window") ;
    printp (20,12,"รหัสค้นพบ: ",NM_HL_VDO) ;
    strcpy (buff,res_buf.stu_id) ;
    row = res_var[0].row ;
    col = res_var[0].col ;
    res_var[0].row = 20 ;
    res_var[0].col = 21 ;
    new = NEW ;
    do

```

```
{    status = fld_edit (new,&res_var[0],NAUTO_ADV,NULL,NULL) ;
    new = OLD ;
} while (status != RETURN && status != F10 && status != ESC) ;
res_var[0].row = row    ; res_var[0].col = col ;
recall () ;
if (status == ESC)
{
    strcpy (res_buf.stu_id,buff) ;
    return ;
}
strcpy (res_key0.stu_id,res_buf.stu_id) ;
STATUS = BTRV (B_GETEQ,res_blk,&res_rec,&res_len,&res_key0,0) ;
if (STATUS != 0)
{
    b_error ("ไม่พบรหัสนักศึกษาในแฟ้มข้อมูล") ;
    return ;
}
resd_buff () ;
scr_mgr (OLD,DISP_FLAG,OLD,res_var,res_lbl,9,NULL,NULL) ;
resd_name () ;
}
```



```

strcpy(subj_rec.subj_name,subj_buf.subj_name);
strcpy(subj_rec.subj_ename,subj_buf.subj_ename);
subj_rec.subj_cred = subj_buf.subj_cred;
subj_rec.subj_theory = subj_buf.subj_theory;
subj_rec.subj_opr = subj_buf.subj_opr;
strcpy(subj_rec.subj_prev1,subj_buf.subj_prev1);
strcpy(subj_rec.subj_prev2,subj_buf.subj_prev2);
strcpy(subj_rec.subj_prev3,subj_buf.subj_prev3);
}

subj_menu()
{
int i ;
static struct lbl_template sub_menu[] = {
    {23,4,"1:ถอน",RV_VDO},
    {23,20,"2:เพิ่ม",RV_VDO},
    {23,35,"3:ค้นหา",RV_VDO},
    {23,53,"9:EXIT",RV_VDO}
};

i = get_func(F3,sub_menu);
return(i);
}

subj_prev()
{
STATUS=BTRV(B_GETPR,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
if(STATUS !=0)
{
b_error("กองข้อมูลแรกแล้วครับ");
return;
}
subj_copy_buf();
scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_next()
{
STATUS=BTRV(B_GETNX,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
if(STATUS !=0)
{
b_error("กองข้อมูลท้ายแล้วครับ");
return;
}
subj_copy_buf();
scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_srch()
{
int status,new,row,col;
char buff[7];

```

```
make_win(18,6,22,36,"Search Window");
printp(20,9,"รหัสวิชา",NM_HL_VDO);
strcpy(buff,subj_buf.subj_code);
row = subj_var[0].row ;
col =subj_var[0].col;
subj_var[0].row =20;
subj_var[0].col=21;
new = NEW;
do
{
status = fld_edit(new,&subj_var[0],NAUTO_ADV,NULL,NULL);
new = OLD;
} while(status != RETURN && status !=F10 && status !=ESC);
subj_var[0].row = row ;
subj_var[0].col = col;
recall();
if(status ==ESC)
{
strcpy(subj_buf.subj_code,buff);
return;
}
strcpy(subj_key0.subj_code,subj_buf.subj_code);
STATUS=BTRV(B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
if(STATUS != 0)
{
b_error("ไม่พบรหัสวิชาในแฟ้มข้อมูล");
return;
}
subj_copy_buf();
scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}
```

```

#include "extern.inc"
#include "stue.inc"

re_sume ()
{
    int    func_no ;
    pre_stu() ;
    stu_scr () ;

    scr_mgr (NEW,DISP_FLAG,NEW,stu_var,stu_lbl,18,NULL,NULL) ;
    if (STATUS == 0)
    {
        cpy_stu () ;
        scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
        stu_print () ;
        stu_date1 () ;
        stu_date2 () ;
    }
    do
    {
        switch (func_no=stu_menu())
        {
            case F1 :
                nothing () ;
                break ;
        }
    } while (func_no != F9) ;
}

pre_stu()
{
    static struct var_template  taro_var[] = {
        {15,40,STRING,"99-999999","99-999999",}
    };

    static struct lbl_template  taro_lbl[] = {
        {15,29,"รหัส นศ.ศึกษา :",NM_VDO}
    };

    int    status,new,row,col ;
    char   buff[10] ;

    clrscr() ;
    taro_var[0].datatag.str_add = buff ;
    make_win(13,25,17,53," ปณนร.สน.ภคศึกษา ") ;
    printp (15,29,"รหัส นศ.ศึกษา",NM_VDO) ;
    scr_mgr (NEW,DISP_FLAG,NEW,taro_var,taro_lbl,1,NULL,NULL) ;

    T_E_flag = Eng ;
    cur_fld = 0 ;

    strcpy(buff,stu_buf.stu_id);
    row = stu_var[0].row;
    col = stu_var[0].col;
    stu_var[0].row = 15;
    stu_var[0].col = 40;

```

```

new = NEW;
do
{
    status = fld_edit(new,&stu_var[0],NAUTO_ADV,NULL,NULL) ;
    new = OLD;
}
while (status != RETURN && status != F10 && status !=ESC);
stu_var[0].row = row ;
stu_var[0].col = col;
recall();
if (status == ESC)
{
    strcpy(stu_buf.stu_id,buff);
    return;
}
strcpy(stu_key0.id,stu_buf.stu_id);
STATUS = BTRV(B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0);
if (STATUS == 0)
{
    cpy_stu () ;
    scr_mgr (OLD,DISP_FLAG,OLD,stu_var,stu_lbl,18,NULL,NULL) ;
    stu_print () ;
    stu_date1 () ;
    stu_date2 () ;
    return ;
}
else
    b_error ("ไม่พบรหัสนักศึกษาในงานแฟ้มของมจร") ;
return ;
}

stu_cpy ()
{
    strcpy (stu_rec.stu_id,stu_buf.stu_id) ;
    stu_rec.part = stu_buf.part;
    strcpy (stu_rec.th_name,stu_buf.th_name) ;
    stu_rec.sex = stu_buf.sex ;
    strcpy (stu_rec.th_sname,stu_buf.th_sname) ;
    strcpy (stu_rec.ename,stu_buf.ename) ;
    strcpy (stu_rec.e_sname,stu_buf.e_sname) ;
    strcpy (stu_rec.d_birth,stu_buf.d_birth) ;
    strcpy (stu_rec.d_cont,stu_buf.d_cont) ;
    strcpy (stu_rec.adv_name,stu_buf.adv_name);
    strcpy (stu_rec.add1,stu_buf.add1) ;
    strcpy (stu_rec.add2,stu_buf.add2) ;
    strcpy (stu_rec.tel,stu_buf.tel) ;
    strcpy (stu_rec.p_name,stu_buf.p_name) ;
    strcpy (stu_rec.p_sname,stu_buf.p_sname) ;
    strcpy (stu_rec.p_add1,stu_buf.p_add1) ;
    strcpy (stu_rec.p_add2,stu_buf.p_add2) ;
    strcpy (stu_rec.p_tel,stu_buf.p_tel) ;
}

```



```

{
    strcpy (stu_buf.stu_id,stu_rec.stu_id) ;
    stu_buf.part = stu_rec.part;
    strcpy (stu_buf.th_name,stu_rec.th_name) ;
    stu_buf.sex = stu_rec.sex ;
    strcpy (stu_buf.th_sname,stu_rec.th_sname) ;
    strcpy (stu_buf.ename,stu_rec.ename) ;
    strcpy (stu_buf.e_sname,stu_rec.e_sname) ;
    strcpy (stu_buf.d_birth,stu_rec.d_birth) ;
    strcpy (stu_buf.d_cont,stu_rec.d_cont) ;
    strcpy (stu_buf.adv_name,stu_rec.adv_name);
    strcpy (stu_buf.add1,stu_rec.add1) ;
    strcpy (stu_buf.add2,stu_rec.add2) ;
    strcpy (stu_buf.tel,stu_rec.tel) ;
    strcpy (stu_buf.p_name,stu_rec.p_name) ;
    strcpy (stu_buf.p_sname,stu_rec.p_sname) ;
    strcpy (stu_buf.p_add1,stu_rec.p_add1) ;
    strcpy (stu_buf.p_add2,stu_rec.p_add2) ;
    strcpy (stu_buf.p_tel,stu_rec.p_tel) ;
}

stu_scr ()
{
    clrscr () ;
    frame ( 1, 2,21,80,"ประมวลผลงานค.วน.ภค.กษา" ) ;
    printp ( 9,39,"[dd/mm/yy]",NM_VDO) ;
    printp ( 10,39,"[dd/mm/yy]",NM_VDO) ;
}

stu_menu ()
{
    int i;
    static struct lbl_template fu_stu[] = {
        {23,18,"",NM_VDO},
        {23,53,"9:Exit",RV_VDO}
    };

    i = get_func (F1,fu_stu) ;
    return (i) ;
}

stu_fld (stat)
int *stat ;
{
    if (*stat == ESC)
        return ;
    if (old_fld == 1 || *stat == F10 )
    {
        if (stu_buf.part > 4 )
        {
            error("โปรดระบุ ทศ.วน.ภค.กษา");
            cur_fld = old_fld;
            *stat = RETURN ;
            return;
        }
    }
}

```

```

    }
}
if (old_fld == 3 || *stat == F10)
{
    if (stu_buf.sex > 1)
    {
        error ("โปรดระวังเพศ");
        cur_fld = old_fld ;
        *stat = RETURN ;
        return;
    }
}
if (old_fld == 7 || *stat == F10)
{
    if (chk_date(stu_var[7].datatag.str_add) == FALSE)
    {
        printp ( 9, 53, "                                ", NM_VDO)

        error ("วันหมดเขตผลาติ..");
        cur_fld = old_fld ;
        *stat = RETURN ;
    }
    else
    {
        stu_date1 () ;
    }
    return;
}
if (old_fld == 8 || *stat == F10)
{
    if (chk_date(stu_var[8].datatag.str_add) == FALSE)
    {
        printp ( 10, 53, "                                ", NM_VDO)

        error ("วันหมดเขตผลาติ..");
        cur_fld = old_fld ;
        *stat = RETURN ;
    }

    else
    {
        stu_date2 () ;
    }

    return;
}
}

stu_date1 ()
{

```

เอกสารนี้เป็นเอกสารที่ char ไว้สื่อบูฟ[25]; ซึ่งงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (chk_date(stu_var[7].datatag.str_add)==FALSE)
            return ;
        buf[0] = '\0' ;
        tsci_date (stu_var[7].datatag.str_add,buf) ;
        printp ( 9,53,"",NM_VDO) ;
        printp ( 9,53,buf,NM_HL_VDO) ;
    }

stu_date2 ()
{
    char buf[25];

    if (chk_date(stu_var[8].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (stu_var[8].datatag.str_add,buf) ;
    printp ( 10,53,"",NM_VDO) ;
    printp ( 10,53,buf,NM_HL_VDO) ;
}

stu_dep(cod,buf)
int cod;
char buff[];
{
    static char *dep [] = { "คณะศาสตร์และวิทยาการคอมพิวเตอร์","เคมี",
        "ชีววิทยาประยุกต์", "ฟิสิกส์ประยุกต์",
        "สังคมประยุกต์" };
    strcpy(buff,dep[cod]);
}

stu_print ()
{
    char buf1[5] ;
    char buf2[41];
    stu_dep (stu_buf.part,buf2);
    printp (stu_var[1].row,stu_var[1].col,buf2,NM_HL_VDO) ;
    printp (4,20,"",NM_HL_VDO) ;

    printp (4,20,buf2,NM_HL_VDO);
    stu_sex (stu_buf.sex,buf1) ;
    printp (stu_var[3].row,stu_var[3].col,buf1,NM_HL_VDO) ;
    printp (5,55,"",NM_HL_VDO);
    printp (5,55,buf1,NM_HL_VDO);
}

stu_sex (code,buf)
int code ;
char buf[] ;
{
    static char *sex[] = {"ชาย",
        "หญิง"} ;
    strcpy (buf,sex[code]);
}

```

```
}  
nothing()  
{  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include "extern.inc"
#include "stue.inc"

list_sub()
{
    int func_no;

    subj_scr();
    scr_mgr(NEW,DISP_FLAG,NEW,sub_j_var,subj_lbl,9,NULL,NULL);
    STATUS=BTRV(B_GETLW,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if(STATUS ==0)
    {
        subj_copy_buf();
        scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
    }
    do {
        switch(func_no =subj_menu())
        {
            case F1:
                subj_prev();
                break;
            case F2:
                subj_next();
                break;
            case F3:
                subj_srch();
                break;
            case F4:
                subj_edit();
                break;
            case F5:
                subj_inc();
                break;
            case F6:
                subj_del();
                break;
        }
    } while (func_no !=F9);
}

subj_scr()
{
    clrscr();
    frame(3,1,16,80,"รายชื่อของหน่วยงานในหลักสูตร");
    printp(11,22,"-",NM_HL_VDO);
    printp(11,31,"-",NM_HL_VDO);
    printp(11,40,"-",NM_HL_VDO);
    printp(24,10,">",NM_HL_VDO);
    printp(25,10,"<",NM_HL_VDO);
}

subj_copy_buf()
{
    strcpy(subj_buf.subj_code,subj_rec.subj_code);
    strcpy(subj_buf.subj_name,subj_rec.subj_name);
}
```

```

strcpy(subj_buf.subj_ename,subj_rec.subj_ename);
subj_buf.subj_cred = subj_rec.subj_cred;
subj_buf.subj_theory = subj_rec.subj_theory;
subj_buf.subj_opr = subj_rec.subj_opr;
strcpy(subj_buf.subj_prev1,subj_rec.subj_prev1);
strcpy(subj_buf.subj_prev2,subj_rec.subj_prev2);
strcpy(subj_buf.subj_prev3,subj_rec.subj_prev3);
}

subj_copy_rec()
{
    strcpy(subj_rec.subj_code,subj_buf.subj_code);
    strcpy(subj_rec.subj_name,subj_buf.subj_name);
    strcpy(subj_rec.subj_ename,subj_buf.subj_ename);
    subj_rec.subj_cred = subj_buf.subj_cred;
    subj_rec.subj_theory = subj_buf.subj_theory;
    subj_rec.subj_opr = subj_buf.subj_opr;
    strcpy(subj_rec.subj_prev1,subj_buf.subj_prev1);
    strcpy(subj_rec.subj_prev2,subj_buf.subj_prev2);
    strcpy(subj_rec.subj_prev3,subj_buf.subj_prev3);
}

subj_menu()
{
    static struct lbl_template    sub_menu[] = {
        {23,4,"1:ก.อน",RV_VDO},
        {23,11,"2:พล.ง",RV_VDO},
        {23,18,"3:ค.พห",RV_VDO},
        {23,26,"4:ก.ช",RV_VDO},
        {23,35,"5:พ.ม.ต.ม",RV_VDO},
        {23,46,"6:ลบ",RV_VDO},
        {23,71,"9:EXIT",RV_VDO}
    };

    int    i ;

    i = get_func(F6,sub_menu);
    return (i);
}

subj_prev()
{
    STATUS=BTRV(B_GETPR,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if(STATUS !=0)
    {
        b_error("ก.ง.ข.อม.ง.ล.ร.ก.แล.ง.ว.ค.ร.บ.");
        return;
    }
    subj_copy_buf();
    scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_next()
{
    STATUS=BTRV(B_GETNX,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if(STATUS !=0)

```



```

    {
        b_error("การแก้ไขข้อมูลภายในผิดพลาด");
        return;
    }
    subj_copy_buf();
    scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_inc()
{
    int status;

    cur_fld = 0;
    scr_mgr(NEW,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
    T_E_flag = Eng;
    do {
        switch (cur_fld)
        {
            case 1:
                T_E_flag = Thai;
                break;
            default :
                T_E_flag = Eng;
                break;
        }
        status = scr_mgr(OLD,ED_FLAG,OLD,subj_var,subj_lbl,9,NULL,N
ULL);
    } while (status != F10 && status !=ESC);
    if (status == ESC)
    {
        subj_copy_buf();
        scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
        return ;
    }
    subj_copy_rec();
    instr (0,"การแก้ไขข้อมูล...");
    STATUS = BTRV (B_INSERT,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if (STATUS != 0)
    {
        b_error("ไม่สามารถแก้ไขข้อมูลเข้าแฟ้มได้");
        STATUS=BTRV(B_GETPR,subj_blk,&subj_rec,&subj_len,&subj_key0
,0);

        if(STATUS !=0)
        STATUS=BTRV(B_GETNX,subj_blk,&subj_rec,&subj_len,&subj_key0
,0);

        if(STATUS !=0)
            scr_mgr(NEW,DISP_FLAG,NEW,subj_var,subj_lbl,9,NULL,
NULL);
    }
    scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_edit()
{
    int status;

```

```

cur_fld = 0;
T_E_flag = Eng;
STATUS = BTRV (B_GETEQL,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
do {
    switch (cur_fld)
    {
        case 1:
            T_E_flag = Thai;
            break;
        default:
            T_E_flag = Eng;
            break;
    }
    status = scr_mgr(OLD,ED_FLAG,OLD,subj_var,subj_lbl,9,NULL,N
ULL);
    } while (status != F10 && status != ESC);
    if (status == ESC)
    {
        subj_copy_buf();
        scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
        STATUS=BTRV(B_UNLOCK,subj_blk,&subj_rec,&subj_len,&subj_key
0,0);
        return ;
    }
    subj_copy_rec();
    if(STATUS==0)
        instr (0,"การลบข้อมูล...");
    STATUS=BTRV(B_UPDATE, subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if (STATUS != 0)
    {
        b_error("ไม่สามารถแก้ไขข้อมูลได้");
        STATUS=BTRV(B_GETPR,subj_blk,&subj_rec,&subj_len,&subj_key0
,0);
        if (STATUS != 0)
            STATUS=BTRV(B_GETNX,subj_blk,&subj_rec,&subj_len,&subj_key0
,0);
        if(STATUS !=0)
            scr_mgr(NEW,DISP_FLAG,NEW,subj_var,subj_lbl,9,NULL,
NULL);
    }
    STATUS=BTRV(B_UNLOCK,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}

subj_del()
{
    int sure;

    sure = make_sure ("โปรดยืนยันการลบข้อมูล");
    STATUS=BTRV(B_GETEQL,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if (sure == 0)
        return;
    instr (0,"การลบข้อมูล...");
    STATUS = BTRV (B_DELETE,subj_blk,&subj_rec,&subj_len,&subj_key0,0);

```

```

clrinstr ();
if (STATUS != 0)
{
    b_error("ไม่สามารถลบข้อมูลได้");
    subj_copy_buf();
    scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
    return;
}
STATUS = BTRV (B_GETNX,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
STATUS = BTRV (B_UNLOCK,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
if(STATUS != 0)
{
    STATUS = BTRV (B_GETPR,subj_blk,&subj_rec,&subj_len,&subj_k
ey0,0);
    if (STATUS != 0)
    {
        scr_mgr(NEW,DISP_FLAG,NEW,subj_var,subj_lbl,9,NULL,
NULL);
        return;
    }
    STATUS = BTRV (B_UNLOCK,subj_blk,&subj_rec,&subj_len,&subj_
key0,0);
    subj_copy_buf();
    scr_mgr (OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);
}
}
subj_srch()
{
    int status,new,row,col;
    char buff[7]; /* buuf code */

    T_E_flag = Eng;
    make_win(18,6,22,36,"Search Window");
    printp(20,9,"รห.สาขา",NM_HL_VDO);
    strcpy(buff,subj_buf.subj_code);
    row = subj_var[0].row ;
    col = subj_var[0].col;
    subj_var[0].row = 20;
    subj_var[0].col= 21;
    new = NEW;
    do {
        status = fld_edit(new,&subj_var[0],NAUTO_ADV,NULL,NULL);
        new = OLD;
    } while(status != RETURN && status !=F10 && status !=ESC);
    subj_var[0].row = row ; subj_var[0].col = col;
    recall();
    if (status == ESC)
    {
        strcpy(subj_buf.subj_code,buff);
        return;
    }
    strcpy(subj_key0.subj_code,subj_buf.subj_code);
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
    if(STATUS != 0)

```

```
{  
    b_error("ไม่พบรหัสผ่านแฟ้มข้อมูล");  
    return;  
}  
subj_copy_buf();  
scr_mgr(OLD,DISP_FLAG,OLD,subj_var,subj_lbl,9,NULL,NULL);  
}
```



```

#include "extern.inc"
#include "stue.inc"

open_sub ()
{
    int      func_no ;

    osub_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,osub_var,osub_lbl,6,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS==0)
    {
        osub_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
        osub_print () ;
    }
    do
    {
        switch (func_no = osub_menu())
        {
            case F1:
                osub_prev () ;
                break ;
            case F2:
                osub_next () ;
                break ;
            case F3:
                osub_srch () ;
                break ;
            case F4:
                osub_edit () ;
                break ;
            case F5:
                osub_ins () ;
                break ;
            case F6:
                osub_del () ;
                break ;
        }
    } while (func_no!=F9) ;
}

osub_scr ()
{
    clrscr () ;
    frame (9,5,18,80,"ระบบประมวลผลกระบวนงานชาติแบบอัตโนมัติแบบปัจจุบัน") ;
    printp(12,14,"ชื่ออาสา :",NM_VDO) ;
    printp(13,7,"ชื่ออาสา(อังกฤษ) :",NM_VDO);
    printp(16,29,"[dd/mm]",NM_VDO) ;
}

```

```

strcpy (osub_buf.osub_tea,osub_rec.osub_tea) ;
osub_buf.osub_day = osub_rec.osub_day ;
osub_buf.osub_part1 = osub_rec.osub_part1 ;
strcpy (osub_buf.osub_date,osub_rec.osub_date) ;
osub_buf.osub_part2 = osub_rec.osub_part2 ;
}

osub_buf2rec ()
{
    strcpy (osub_rec.osub_id,osub_buf.osub_id) ;
    strcpy (osub_rec.osub_tea,osub_buf.osub_tea) ;
    osub_rec.osub_day = osub_buf.osub_day ;
    osub_rec.osub_part1 = osub_buf.osub_part1 ;
    strcpy (osub_rec.osub_date,osub_buf.osub_date) ;
    osub_rec.osub_part2 = osub_buf.osub_part2 ;
}

osub_print ()
{
    osub_name () ;
    select_date () ;
    osub_part (osub_buf.osub_part1,15,58) ;
    osub_date (osub_buf.osub_date,16,38) ;
    osub_part (osub_buf.osub_part2,16,58) ;
}

osub_name ()
{
    char nsub[61] ;
    char ensub[61] ;

    strcpy (subj_key0.subj_code,osub_buf.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS!=0)
    {
        bt_error () ;
        return (FALSE) ;
    }
    strcpy (nsub,subj_rec.subj_name) ;
    strcpy (ensub,subj_rec.subj_ename) ;
    nsub[49] = '\0' ;
    ensub[55] = '\0' ;
    printp (12,22,"",NM_VD
0) ;
    printp (12,22,nsub,NM_HL_VDO) ;
    printp (13,22,"
,NM_VDO) ;
    printp (13,22,ensub,NM_HL_VDO) ;
    return (TRUE) ;
}

select_date ()
{
    static char *day[] = {"จ.นทร์",

```



```

        "อังคาร",
        "พุธ",
        "พฤหัสบดี",
        "ศุกร์",
        "เสาร์",
        "อาทิตย์");

char    dbuf[9] ;

strcpy (dbuf,day[osub_buf.osub_day]) ;
printp (15,25,"          ",NM_VDO) ;
printp (15,25,dbuf,NM_HL_VDO) ;
}

osub_date (dm,a,b)
char    dm[] ;
int     a, b ;
{
static char    *osub_mon[] = { "ม.ค.", "ก.พ.", "มี.ค.", "เม.ย.", "พ.ค.", "มิ.ย.",
                                "ก.ค.", "ส.ค.", "ก.ย.", "ต.ค.", "พ.ย.", "ธ.ค." }
;

char    buf1[9], buf2[6] ;
int     mon ;

if (osub_chk(dm)==FALSE)
{
    error ("ว.นพ.ผิด") ;
    return (FALSE) ;
}
strcpy (buf2, dm) ;
buf1[0] = '\0' ;
mon = (buf2[3]-'0')*10 + buf2[4]-'0' ;
buf2[2] = '\0' ;
strcat (buf1, buf2) ;
strcat (buf1, " ") ;
strcat (buf1, osub_mon[mon-1]) ;
if (buf1[0]!='0')
    buf1[0] = ' ' ;
printp (a,b,"          ",NM_VDO) ;
printp (a,b,buf1,NM_HL_VDO) ;
return (TRUE) ;
}

osub_chk (s)
char    *s ;
{
    int     d, m, i ;
    int     dinm[12] = {31,29,31,30,31,30,31,31,30,31,30,31} ;

    for (i=0; i<6; i+=3)
    {
        if (s[i] == ' ')
            s[i] = '0' ;
        if (s[i+1] == ' ' && s[i] >= '0' && s[i] <= '9')
            {

```

```

        s[i+1] = s[i] ;
        s[i] = '0' ;
    }
}
d = (s[0]-'0')*10 + s[1]-'0' ;
m = (s[3]-'0')*10 + s[4]-'0' ;
if (m < 1 && m > 12)
    return (FALSE) ;
if (d > dinm[m-1] || d < 1)
    return (FALSE) ;
else
    return (TRUE) ;
}

osub_part (pa,a,b)
int    pa,a,b ;
{
    static char    *part[] = { "๕๖๗", "๘๙๐" } ;

    printp (a,b,"      ",NM_VDO) ;
    printp (a,b,part[pa],NM_HL_VDO) ;
}

osub_menu ()
{
    static struct lbl_template fn_osub[] = { {23, 4,"1:กดปุ่ม",RV_VDO},
                                              {23,11,"2:หล่น",RV_VDO},
                                              {23,18,"3:ค้นหา",RV_VDO},
                                              {23,26,"4:แก้ไข",RV_VDO},
                                              {23,35,"5:เพิ่มข้อมูล",RV_VDO},
                                              {23,46,"6:ลบ",RV_VDO},
                                              {23,71,"9:Exit",RV_VDO}
                                              } ;

    int    i ;

    i = get_func (F6,fn_osub) ;
    return (i) ;
}

osub_prev ()
{
    STATUS = BTRV (B_GETPR,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("กดปุ่มมวงลแรกแล้วครับ") ;
        return ;
    }
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
}

osub_next ()
{
    STATUS = BTRV (B_GETNX,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;

```

```

    if (STATUS != 0)
    {
        b_error ("การแก้ไขข้อมูลผิดพลาด");
        return ;
    }
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
}

osub_srch ()
{
    int      status, new, row, col ;
    char     buff[10] ;

    T_E_flag = Eng ;
    make_win (18, 6,22,36,"Search Window") ;
    printp (20,12,"รหัสผ่าน : ",NM_HL_VDO) ;
    strcpy (buff,osub_buf.osub_id) ;
    row = osub_var[0].row ; col = osub_var[0].col ;
    osub_var[0].row = 20 ; osub_var[0].col = 21 ;
    new = NEW ;
    do
    {
        status = fld_edit (new,&osub_var[0],NAUTO_ADV,NULL,NULL) ;
        new = OLD ;
    } while (status != RETURN && status != F10 && status != ESC) ;
    osub_var[0].row = row ; osub_var[0].col = col ;
    recall () ;
    if (status == ESC)
    {
        strcpy (osub_buf.osub_id,buff) ;
        return ;
    }
    strcpy (osub_key0.osub_id,osub_buf.osub_id) ;

    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบรหัสผ่านในแฟ้มข้อมูล") ;
        return ;
    }
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
}

osub_edit ()
{
    int      status ;

    cur_fld = 0 ;
    T_E_flag = Eng ;
    STATUS=BTRV(B_GETEQL,osub_blk,&osub_rec,&osub_len,&osub_key0,0);
    do
    {
        switch (cur_fld)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีโทษทางกฎหมายและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{      case 1 :
            T_E_flag = Thai ;
            break ;
        default : T_E_flag = Eng ;
            break ;
    }
    if (cur_fld == 2)
    {
        status = fld_sel (&ssdate_var[0],&date_selv[0],date_selv[0]
.svar_struct,NULL,NULL) ;
        if (status != ESC)
            osub_buf.osub_day = date_selv[0].cur_sv ;
        if (status == F10 || status == ESC)
            break ;
        cur_fld = 3 ;
    }
    if (cur_fld == 3)
    {
        status = fld_sel (&sspa1_var[0],&pa_selv[0],pa_selv
[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            osub_buf.osub_part1 = pa_selv[0].cur_sv ;
        if (status==F10 || status==ESC)
            break ;
        cur_fld = 4 ;
    }
    if (cur_fld == 5)
    {
        status = fld_sel (&sspa2_var[0],&pa_selv[0],pa_selv
[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            osub_buf.osub_part2 = pa_selv[0].cur_sv ;
        if (status==F10 || status==ESC)
            break ;
        cur_fld = 0 ;
    }
    status = scr_mgr (OLD,ED_FLAG,OLD,osub_var,osub_lbl,6,NULL,
NULL) ;

    osub_fld (&status) ;
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    STATUS=BTRV(B_UNLOCK,osub_blk,&osub_rec,&osub_len,&osub_key
0,0);

    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
    return ;
}
osub_buf2rec () ;
instr (0,"การแก้ไขข้อมูล...") ;
STATUS = BTRV (B_UPDATE,osub_blk,&osub_rec,&osub_len,&osub_key0,0)

```

```

if (STATUS != 0)
{
    b_error ("ไม่สามารถเก็บเข้าแฟ้มข้อมูลได้");
    STATUS = BTRV (B_GETGE,osub_blk,&osub_rec,&osub_len,&osub_key0,0);
    if (STATUS != 0)
        scr_mgr (NEW,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL
, NULL) ;
    else
    {
        osub_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL
, NULL) ;
        osub_print () ;
    }
    return ;
}
STATUS=BTRV(B_UNLOCK,osub_blk,&osub_rec,&osub_len,&osub_key0,0);
scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
osub_print () ;
}
osub_ins ()
{
    int    status ;

    cur_fld = 0 ;
    scr_mgr (NEW,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    printp (12,22,"",NM_VDO) ;
    printp (13,22,"",NM_VDO) ;
    printp (15,25,"",NM_VDO) ;
    printp (15,58,"",NM_VDO) ;
    printp (16,38,"",NM_VDO) ;
    printp (16,58,"",NM_VDO) ;
    T_E_flag = Eng ;
    do
    {
        switch (cur_fld)
        {
            case 1 :
                T_E_flag = Thai ;
                break ;
            default : T_E_flag = Eng ;
                break ;
        }

        if (cur_fld == 2)
        {
            status = fld_sel (&ssdate_var[0],&date_selv[0],date_selv[0]
.svar_struct,NULL,NULL) ;
            if (status != ESC)
                osub_buf.osub_day = date_selv[0].cur_sv ;
            if (status == F10 || status == ESC)
                break ;
            cur_fld = 3;
        }
    }
}

```

```

    }

    if (cur_fld == 3)
    {
        status = fld_sel (&sspa1_var[0],&pa_selv[0],pa_selv
[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            osub_buf.osub_part1 = pa_selv[0].cur_sv ;
        cur_fld = 4 ;
        if (status==F10 || status==ESC)
            break ;
    }
    if (cur_fld == 5)
    {
        status = fld_sel (&sspa2_var[0],&pa_selv[0],pa_selv
[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            osub_buf.osub_part2 = pa_selv[0].cur_sv ;
        if (status==F10 || status==ESC)
            break ;
        cur_fld = 0 ;
    }
    status = scr_mgr (OLD,ED_FLAG,OLD,osub_var,osub_lbl,6,NULL,
NULL) ;
    osub_fld (&status) ;
} while (status != F10 && status != ESC) ;
if (status == ESC)
{
    osub_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
    osub_print () ;
    return ;
}
osub_buf2rec () ;
instr (0,"การแก้ไขข้อมูล....") ;
STATUS = BTRV (B_INSERT,osub_blk,&osub_rec,&osub_len,&osub_key0,0)
;

if (STATUS != 0)
{
    b_error ("ไม่สามารถแก้ไขแฟ้มข้อมูลได้") ;
    STATUS = BTRV (B_GETPR,osub_blk,&osub_rec,&osub_len,&osub_k
ey0,0) ;

    if (STATUS!=0)
        STATUS = BTRV (B_GETNX,osub_blk,&osub_rec,&osub_len
,&osub_key0,0) ;
    if (STATUS!=0)
        scr_mgr (NEW,DISP_FLAG,NEW,osub_var,osub_lbl,6,NULL
,NULL) ;
}
scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL) ;
osub_print () ;
}

osub_del ()

```



```

int    sure ;

sure = make_sure ("โปรดหยุดการลบข้อมูล") ;
if (sure == 0)
    return ;
STATUS=BTRV(B_GETEQL,subj_blk,&subj_rec,&subj_len,&subj_key0,0);
instr ( 0,"การลบข้อมูล..." ) ;
STATUS = BTRV ( B_DELETE,osub_blk,&osub_rec,&osub_len,&osub_key0,0
) ;

clrinstr () ;
if ( STATUS != 0 )
{
    scr_mgr ( OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL )
;

    osub_print () ;
    return ;
}
STATUS=BTRV(B_UNLOCK,osub_blk,&osub_rec,&osub_len,&osub_key0,0);
STATUS = BTRV ( B_GETNX,osub_blk,&osub_rec,&osub_len,&osub_key0,0 )
;

if (STATUS != 0)
STATUS = BTRV( B_GETPR,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
if (STATUS != 0)
{
    scr_mgr (NEW,DISP_FLAG,NEW,osub_var,osub_lbl,6,NULL,NULL) ;
    return ;
}
osub_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,osub_var,osub_lbl,6,NULL,NULL ) ;
osub_print () ;
}

osub_fld (st)
int    *st ;
{
    if (*st==ESC)
        return ;
    if (old_fld==0 && cur_fld!=0)
    {
        if (osub_name ()==FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
        return ;
    }
    if (old_fld == 2 && cur_fld >2)
    {
        if (select_date ()==FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        return ;
    }
    if (old_fld == 4 && cur_fld > 4)
    {
        if (osub_date (osub_buf.osub_date,16,38)==FALSE)
        {
            cur_fld = old_fld ;
            *st = RETURN ;
            return ;
        }
        return ;
    }
}
```



```

#include "extern.inc"
#include "stue.inc"

plan_edu ()
{
    int    func_no ;

    plan_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,plan_var,plan_lbl,15,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
    if (STATUS == 0)
    {
        plan_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL)
;

        plan_print () ;
    }
    plan_clr0 () ;
    do {
        switch (func_no=plan_menu())
        {
            case F1:
                plan_prev () ;
                break ;
            case F2:
                plan_next () ;
                break ;
            case F3:
                plan_edit () ;
                break ;
            case F4:
                plan_ins () ;
                break ;
            case F5:
                plan_del () ;
                break ;
        }
    } while (func_no != F9) ;
}

plan_scr ()
{
    clrscr () ;
    frame ( '3, 1,20,79,"โครงการการศึกษา" ) ;
    lineH (1,79,6) ;
    lineV (6,20,10) ;
    lineV (6,20,58) ;
    lineV (6,20,67) ;
    printp (6,3,"รหัสวิชา",RV_VDO) ;
    printp (6,30,"ชื่อวิชา",RV_VDO) ;
    printp (6,60,"หน่วยกิต",RV_VDO) ;
    printp (6,69,"ทฤษฎี-ปฏิบัติ",RV_VDO) ;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 plan_rec2buf ()
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    int    i ;

    plan_buf.plan_dep = plan_rec.plan_dep ;
    plan_buf.plan_sem = plan_rec.plan_sem ;
    plan_buf.plan_year = plan_rec.plan_year ;
    for (i=0; i<12; i++)
        strcpy (plan_buf.sub[i].code,plan_rec.sub[i].code) ;
}

plan_buf2rec ()
{
    int    i ;

    plan_rec.plan_dep = plan_buf.plan_dep ;
    plan_rec.plan_sem = plan_buf.plan_sem ;
    plan_rec.plan_year = plan_buf.plan_year ;
    for (i=0; i<12; i++)
    {
        plan_buf.sub[i].code[6] = '\0' ;
        strcpy (plan_rec.sub[i].code,plan_buf.sub[i].code) ;
    }
}

plan_menu ()
{
    static struct lbl_template fn_plan[] = {{23, 4,"1:ก.อน",RV_VDO},
                                             {23,11,"2:พล.ง",RV_VDO},
                                             {23,18,"3:เกษ.ย",RV_VDO},
                                             {23,26,"4:พ.อ.ม.เต.ม",RV_VDO},
                                             {23,35,"5:ลพ",RV_VDO},
                                             {23,71,"9:Exit",RV_VDO}
                                             };

    int    i ;

    i = get_func (F5,fn_plan) ;
    return (i) ;
}

plan_print ()
{
    int    i ;

    plan_dep () ;
    plan_sem () ;
    for (i=0; i<12; i++)
    {
        if (plan_buf.sub[i].code[0] != ' ')
            plan_getsub (i) ;
    }
}

plan_clr0 ()

```



```

        "เทคโนโลยีชีวภาพ",
        "พาสถิกรประย กตศ",
        "สถิตยประย กตศ"} ;

char    dbuf[41] ;

strcpy (dbuf,dep[plan_buf.plan_dep]) ;
printp (4,17,"                                ",NM_VDO) ;
printp (4,17,dbuf,NM_HL_VDO) ;
}

plan_sem ()
{
static char    *sem[] = {"1st",
                          "2nd",
                          "Summer" } ;

char    sbuf[7] ;

strcpy (sbuf,sem[plan_buf.plan_sem]) ;
printp ( 5,17,"                                ",NM_VDO) ;
printp ( 5,17,sbuf,NM_HL_VDO) ;
}

plan_prev ()
{
STATUS = BTRV (B_GETPR,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
if (STATUS != 0)
{
    b_error ("กองบงคมยลแรกแลจวศร.บ") ;
    return ;
}
plan_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
plan_print () ;
plan_clr0 () ;
}

plan_next ()
{
STATUS = BTRV (B_GETNX,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
if (STATUS != 0)
{
    b_error ("กองบงคมยลศ ิตททยแลจวศร.บ") ;
    return ;
}
plan_rec2buf () ;
scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
plan_print () ;
plan_clr0 () ;
}

plan_edit ()
{
int    status, st ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

cur_fld = 0 ;
do {
    if (cur_fld==0)
    {
        status = fld_sel (&pldep_var[0],&dep_selv[0],dep_selv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            plan_buf.plan_dep = dep_selv[0].cur_sv ;
        if (status == F10 || status == ESC)
            break ;
        cur_fld = 1 ;
    }
    if (cur_fld==1)
    {
        status = fld_sel (&plsem_var[0],&ssem_selv[0],ssem_selv[0].svar_struct,NULL,NULL) ;
        if (status != ESC)
            plan_buf.plan_sem = ssem_selv[0].cur_sv ;
        if (status == F10 || status == ESC)
            break ;
        if (status==CUR_UP)
            cur_fld = 0 ;
        else
            cur_fld = 2 ;
    }
    if (cur_fld > 3 && old_fld > 2)
    {
        st = plan_getsub(old_fld-3) ;
        if (st == FALSE)
            cur_fld = old_fld ;
    }
    if (cur_fld != 0)
        status = scr_mgr (OLD,ED_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
    } while (status != F10 && status != ESC) ;
    if (status == ESC)
    {
        plan_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
        plan_print () ;
        plan_clr0 () ;
        return ;
    }
    instr ("การลบแก้ไขข้อมูล....") ;
    STATUS = BTRV (B_DELETE,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
    plan_sort () ;
    plan_buf2rec () ;
    STATUS = BTRV (B_INSERT,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
    if (STATUS != 0)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ey0,0) ;

        if (STATUS != 0)
            scr_mgr (NEW,DISP_FLAG,OLD,plan_var,plan_lbl,15,NUL
L,NULL) ;
        else
        {
            plan_rec2buf () ;
            scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NUL
L,NULL) ;

            plan_print () ;
            plan_clr0 () ;
        }
        return ;
    }
    scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
    plan_print () ;
    plan_clr0 () ;
}

plan_ins ()
{
    int      status, st ;

    scr_mgr (NEW,DISP_FLAG,NEW,plan_var,plan_lbl,15,NULL,NULL) ;
    clr_win (7,11,19,56) ;
    clr_win (7,60,19,66) ;
    clr_win (7,70,19,76) ;
    printp (4,17,"
";
    printp (5,17,"
",NM_VDO) ;
    T_E_flag = Eng ;
    cur_fld = 0 ;
    do {
        if (cur_fld==0)
        {
            status = fld_sel (&pldep_var[0],&dep_selv[0],dep_se
lv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                plan_buf.plan_dep = dep_selv[0].cur_sv ;
            if (status == F10 || status == ESC)
                break ;
            cur_fld = 1 ;
        }
        if (cur_fld==1)
        {
            status = fld_sel (&plsem_var[0],&ssem_selv[0],ssem_
selv[0].svar_struct,NULL,NULL) ;
            if (status != ESC)
                plan_buf.plan_sem = ssem_selv[0].cur_sv ;
            else
                break ;
            if (status==CUR_UP)
                cur_fld = 0 ;
            else
                cur_fld = 2 ;
        }
    }
}

```

```

    }
    if (cur_fld > 3 && old_fld > 2)
    {
        st = plan_getsub(old_fld-3) ;
        if (st == FALSE)
            cur_fld = old_fld ;
    }
    if (cur_fld != 0)
        status = scr_mgr (OLD,ED_FLAG,OLD,plan_var,plan_lbl
,15,NULL,NULL) ;
    } while (status != F10 && status != ESC) ;
    if (status == ESC)
    {
        plan_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL)
;

        plan_print () ;
        plan_clr0 () ;
        return ;
    }
    instr ("กําลังลบข้อมูล....") ;
    plan_sort () ;
    plan_buf2rec () ;
    STATUS = BTRV (B_INSERT,plan_blk,&plan_rec,&plan_len,&plan_key0,0)
;
    if (STATUS != 0)
    {
        STATUS = BTRV (B_GETPR,plan_blk,&plan_rec,&plan_len,&plan_k
ey0,0) ;
        if (STATUS != 0)
        STATUS = BTRV (B_GETNX,plan_blk,&plan_rec,&plan_len,&plan_k
ey0,0) ;
        if (STATUS != 0)
        {
            clr_win (4,17,4,70) ;
            clr_win (5,17,5,25) ;
            scr_mgr (NEW,DISP_FLAG,NEW,plan_var,plan_lbl,15,NUL
L,NULL) ;

            plan_clr0 () ;
            return ;
        }
    }
    scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
    plan_print () ;
    plan_clr0 () ;
}

plan_del ()
{
    int    sure ;

    sure = make_sure ("โปรดยืนยันการลบข้อมูล") ;
    if (sure==0)
        return ;
    instr ("กําลังลบข้อมูล....") ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อใช้ในการทำงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

STATUS = BTRV (B_DELETE,plan_blk,&plan_rec,&plan_len,&plan_key0,0)
;
clrinstr ( ) ;
if (STATUS != 0)
{
    b_error ("ไม่สามารถลบข้อมูลได้") ;
    scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL)
;
    plan_print ( ) ;
    plan_clr0 ( ) ;
    return ;
}
STATUS = BTRV (B_GETPR,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
if (STATUS != 0)
STATUS = BTRV (B_GETNX,plan_blk,&plan_rec,&plan_len,&plan_key0,0) ;
if (STATUS != 0)
{
    clr_win (4,17,4,70) ;
    clr_win (5,17,5,25) ;
    scr_mgr (NEW,DISP_FLAG,NEW,plan_var,plan_lbl,15,NULL,NULL)
;
    plan_clr0 ( ) ;
    return ;
}
plan_rec2buf ( ) ;
scr_mgr (OLD,DISP_FLAG,OLD,plan_var,plan_lbl,15,NULL,NULL) ;
plan_print ( ) ;
plan_clr0 ( ) ;
}

plan_sort ( )
{
    char    scode[12][6] ;
    int     tmptmp ;
    int     point1, point2 ;
    int     tmptag[12] ;

    for (point1=0; point1<12; point1++)
    {
        strcpy (scode[point1],plan_buf.sub[point1].code) ;
        tmptag[point1] = point1 ;
    }
    for (point1=0; point1<11; point1++)
    {
        for (point2=point1+1; point2<12; point2++)
        {
            if (strcmp(scode[tmptag[point1]],scode[tmptag[point
2]]) > 0)
            {
                tmptmp = tmptag[point1] ;
                tmptag[point1] = tmptag[point2] ;
                tmptag[point2] = tmptmp ;
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
point1 = 0 ;
while (scode[tmptag[point1]][0] == ' ')
    point1++ ;
while (point1 < 12)
{
    tmptmp = tmptag[11] ;
    for (point2=10; point2 >= 0; point2--)
        tmptag[point2+1] = tmptag[point2] ;
    tmptag[0] = tmptmp ;
    point1++ ;
}
for (point1=0; point1<12; point1++)
    strcpy (plan_buf.sub[point1].code,scode[tmptag[point1]]) ;
}
```



```

#include "extern.inc"
#include "stue.inc"

interval_reg ()
{
    int    func_no ;
    int_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,int_var,int_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,int_blk,&int_rec,&int_len,&int_key0,0) ;
    if (STATUS == 0)
    {
        cpy_int () ;
        scr_mgr (OLD,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NULL) ;
        int_date1 () ;
        int_date2 () ;
        int_date3 () ;
        int_date4 () ;
    }
    do
    {
        switch (func_no=int_menu())
        {
            case F1 :
                int_next () ;
                break ;
            case F2 :
                int_edit () ;
                break ;
            case F3 :
                int_ins () ;
                break ;
        }
    } while (func_no != F9) ;
}

int_cpy ()
{
    strcpy (int_rec.st1,int_buf.st1) ;
    strcpy (int_rec.end1,int_buf.end2) ;
    strcpy (int_rec.st2,int_buf.st2) ;
    strcpy (int_rec.end2,int_buf.end2) ;
}

cpy_int ()
{
    strcpy (int_buf.st1,int_rec.st1) ;
    strcpy (int_buf.end1,int_rec.end1) ;
    strcpy (int_buf.st2,int_rec.st2) ;
    strcpy (int_buf.end2,int_rec.end2) ;
}

int_scr ()

```



```

clrscr () ;
frame ( 5, 5,15,75,"กำหนดช้ เวลาที่จะเบรค" ) ;
printp (6,15,"การล้เวลาเบรค",NM_VDO);
printp (7,35,"[dd/mm/yy]",NM_VDO) ;
printp (8,35,"[dd/mm/yy]",NM_VDO) ;
lineH (5,75,10);
printp (11,15,"การล้เวลาเบรค",NM_VDO);
printp (12,35,"[dd/mm/yy]",NM_VDO) ;
printp (13,35,"[dd/mm/yy]",NM_VDO) ;
}

int_menu ()
{
    int i;
    static struct lbl_template fu_int[] = {
        {23,11,"1:เวลา",RV_VDO},
        {23,26,"2:แก๊ส",RV_VDO},
        {23,35,"3:เวลาเบรค",RV_VDO},
        {23,53,"9:Exit",RV_VDO}
    };

    i = get_func (F3,fu_int) ;
    return (i) ;
}

int_fld (stat)
int *stat ;
{
    if (*stat == ESC)
        return ;
    if (old_fld==0 || *stat == F10 )
    {
        if (chk_date(int_var[0].datatag.str_add)==FALSE)
        {
            printp ( 7,48,"",NM_VDO)

            error ("เวลาเบรคผิดพลาด..") ;
            cur_fld = old_fld ;
            *stat = RETURN ;
        }
        else
            int_datel () ;
        return;
    }
    if (old_fld==1 || *stat == F10 )
    {
        if (chk_date(int_var[1].datatag.str_add)==FALSE)
        {
            printp ( 8,48,"",NM_VDO) ;
            error ("เวลาเบรคผิดพลาด..") ;
            cur_fld = old_fld ;
            *stat = RETURN ;
        }
    }
}

```

```

        else
            int_date2 () ;
        return;
    }
    if (old_fld==2 || *stat == F10 )
    {
        if (chk_date(int_var[2].datatag.str_add)==FALSE)
        {
            printp ( 12,48,"",NM_VDO)

;

            error ("ว.พท. ไม่ติดพลาต..") ;
            cur_fld = old_fld ;
            *stat = RETURN ;
        }
        else
            int_date3 () ;
        return;
    }
    if (old_fld==3 || *stat == F10 )
    {
        if (chk_date(int_var[3].datatag.str_add)==FALSE)
        {
            printp ( 13,48,"",NM_VDO)

;

            error ("ว.พท. ไม่ติดพลาต..") ;
            cur_fld = old_fld ;
            *stat = RETURN ;
        }
        else
            int_date4 () ;
        return;
    }
}

int_datel ()
{
    char    buf[25] ;

    if (chk_date(int_var[0].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (int_var[0].datatag.str_add,buf) ;
    printp (7,48,"",NM_VDO) ;
    printp (7,48,buf,NM_HL_VDO) ;
}

int_date2 ()
{
    char    buf[25];

    if (chk_date(int_var[1].datatag.str_add)==FALSE)
        return ;
    buf[0] = '\0' ;
    tsci_date (int_var[1].datatag.str_add,buf) ;
    printp (8,48,"",NM_VDO) ;

```

```

        printp (8,48,buf,NM_HL_VDO) ;
    }
    int_date3 '('
    {
        char    buf[25] ;

        if (chk_date(int_var[2].datatag.str_add)==FALSE)
            return ;
        buf[0] = '\0' ;
        tsci_date (int_var[2].datatag.str_add,buf) ;
        printp ( 12,48,"                      ",NM_VDO) ;
        printp ( 12,48,buf,NM_HL_VDO) ;
    }

    int_date4 ()
    {
        char    buf[25];

        if (chk_date(int_var[3].datatag.str_add)==FALSE)
            return ;
        buf[0] = '\0' ;
        tsci_date (int_var[3].datatag.str_add,buf) ;
        printp ( 13,48,"                      ",NM_VDO) ;
        printp ( 13,48,buf,NM_HL_VDO) ;
    }

    int_edit '('
    {
        int    status ;

        cur_fld = 0 ;
        T_E_flag = Eng ;
        STATUS=BTRV(B_GETEQL,int_blk,&int_rec,&int_len,&int_key0,0);
        do
        {
            switch (cur_fld)
            {
                default      :
                    T_E_flag = Eng ;
                    break ;
            }
            status = scr_mgr (OLD,ED_FLAG,OLD,int_var,int_lbl,4,NULL,NU
LL) ;

            int_fld (&status) ;
        } while (status != F10 && status != ESC) ;
        if (status == ESC)
        {
            cpy_int () ;
            scr_mgr (OLD,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NULL) ;
            int_date1 () ;
            int_date2 () ;
            int_date3 () ;
            int_date4 () ;
            STATUS=BTRV(B_UNLOCK,int_blk,&int_rec,&int_len,&int_key0,0)
        }
    }

```

```

        return ;
    }
    int_cpy () ;
    instr (0,"การลบข้อมูล....") ;
    STATUS = BTRV (B_UPDATE,int_blk,&int_rec,&int_len,&int_key0,0) ;
    if (STATUS == 0)
    {
        scr_mgr (OLD,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NULL) ;
        int_date1 () ;
        int_date2 () ;
        int_date3 () ;
        int_date4 () ;

        STATUS=BTRV(B_UNLOCK,int_blk,&int_rec,&int_len,&int_key0,0);
    }
    else
        b_error ("ไม่สามารถลบข้อมูลได้") ;
}

int_ins()
{
    int status;

    cur_fld = 0;
    scr_mgr(NEW,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NULL);
    printp(7,48,"",NM_VDO);
    printp(8,48,"",NM_VDO);
    printp(12,48,"",NM_VDO);
    printp(13,48,"",NM_VDO);
    T_E_flag = Eng;
    do
    {
        switch (cur_fld)
        {
            default :
                T_E_flag = Eng ;
                break ;
        }
        status = scr_mgr (OLD,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NU
    LL) ;
        int_fld (&status) ;
    } while (status != F10 && status != ESC) ;
    if (status == ESC)
    {
        cpy_int () ;
        scr_mgr (OLD,DISP_FLAG,OLD,int_var,int_lbl,4,NULL,NULL) ;
        int_date1 () ;
        int_date2 () ;
        int_date3 () ;
        int_date4 () ;
    }
}

```

```

#include "extern.inc"
#include "stue.inc"

selct_subject ()
{
    int    func_no ;

    selct_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,selct_var,selct_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,selct_blk,&selct_rec,&selct_len,&selct_key0,
0) ;
    if (STATUS == 0)
    {
        selct_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,selct_var,selct_lbl,4,NULL,NULL) ;
        selct_print () ;
    }
    selct_clr0 () ;
    do
    {
        switch (func_no = selct_menu())
        {
            case F1 :
                selct_prev () ;
                break;
            case F2 :
                selct_next () ;
                break;
        }
    } while (func_no != F9) ;
}

selct_scr ()
{
    clrscr () ;
    frame (4,1,13,79,"วิชาเลือกเฉพาะภาคการศึกษา") ;
    lineH (1,79,6) ;
    printp (10,12,"ชื่อ_อาชีว : ",NM_VDO) ;
    printp (11,11,"หน่วยกิต : ",NM_VDO) ;
}

selct_rec2buf ()
{
    selct_buf.selct_dep = selct_rec.selct_dep ;
    selct_buf.selct_sem = selct_rec.selct_sem ;
    selct_buf.ksub = selct_rec.ksub ;
    strcpy (selct_buf.slcode,selct_rec.slcode) ;
}

selct_buf2rec ()
{
    selct_rec.selct_dep = selct_buf.selct_dep ;
    selct_rec.selct_sem = selct_buf.selct_sem ;
    selct_rec.ksub = selct_buf.ksub ;

```

```

        strcpy (selct_rec.slcode,selct_buf.slcode) ;
    }

    selct_print ()
    {
        selct_dep () ;
        selct_sem () ;
        selct_slsub () ;
        selct_getsub () ;
    }

    selct_dep ()
    {
        static char *dep[] = {"คณะสัตวศาสตร์และวิทยาการคอมพิวเตอร์",
                                "เคมื่อสัตสากรรม",
                                "เทคโนโลยีชีวภาพ",
                                "ฟสสภสภษษษษษ",
                                "สภสภษษษษษษ" } ;

        char dbuf[41];

        strcpy (dbuf,dep[selct_buf.selct_dep]) ;
        printp (5,12,"",NM_VDO) ;
        printp (5,12,dbuf,NM_HL_VDO) ;
    }

    selct_sem ()
    {
        static char *sem[] = {"1st",
                                "2nd",
                                "Summer" } ;

        char sbuf[7] ;

        strcpy (sbuf,sem[selct_buf.selct_sem]) ;
        printp (5,64,"",NM_VDO) ;
        printp (5,64,sbuf,NM_HL_VDO) ;
    }

    selct_slsub ()
    {
        static char *sub[] = {"กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ",
                                "กล มวษษษษษษษษษษ" } ;

        char sbuf[21];

        strcpy (sbuf,sub[selct_buf.ksub]) ;
        printp (8,20,"",NM_VDO) ;
        printp (8,20,sbuf,NM_HL_VDO) ;
    }

```



```

}

selct_getsub ()
{
    char    gbuf[61] ;
    char    cbuf[2] ;
    int     n ;

    strcpy (osub_key0.osub_id,selct_buf.slcode) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error("ไม่พบการเปิดสอนกระบวนวิชา");
        return (FALSE) ;
    }
    strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("ไม่พบกระบวนวิชาเรียนแผนกของนักศึกษา") ;
        return (FALSE) ;
    }
    strcpy (gbuf,subj_rec.subj_name) ;
    n = 0;
    while (gbuf[n] != ' ' || gbuf[n+1] != ' ')
        n++ ;
    gbuf[n] = '\0' ;
    printp (10,20,"
",NM_VDO) ;
    printp (10,20,gbuf,NM_HL_VDO) ;
    itoa_1 (subj_rec.subj_cred,cbuf,1) ;
    printp (11,20,cbuf,NM_HL_VDO) ;
    return (TRUE) ;
}

selct_clr0 ()
{
    if (selct_buf.slcode[0] == ' ')
    {
        printp (9,20,"
",NM_VDO) ;
        printp (10,20,"
",NM_VDO) ;
        printp (11,20,"
",NM_VDO) ;
    }
}

selct_menu ()
{
    static struct lbl_template fn_selct[] = {{23, 4,"1:ถอน",RV_VDO},
                                              {23,20,"2:ผลต่าง",RV_VDO},
                                              {23,53,"9:Exit",RV_VDO}}

```

```
i = get_func (F2,fn_selct) ;
return (i) ;
}

selct_prev ()
{
    STATUS = BTRV (B_GETPR,selct_blk,&selct_rec,&selct_len,&selct_key0,
0) ;
    if (STATUS != 0)
    {
        b_error ("การขอข้อมูลแรกแล้วเสร็จ") ;
        return ;
    }
    selct_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,selct_var,selct_lbl,4,NULL,NULL) ;
    selct_print () ;
    selct_clr0 () ;
}

selct_next ()
{
    STATUS = BTRV (B_GETNX,selct_blk,&selct_rec,&selct_len,&selct_key0,
0) ;
    if (STATUS != 0)
    {
        b_error ("การขอข้อมูลสุดท้ายแล้วเสร็จ") ;
        return ;
    }
    selct_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,selct_var,selct_lbl,4,NULL,NULL) ;
    selct_print () ;
    selct_clr0 () ;
}
```

```

#include "extern.inc"
#include "stue.inc"

test_sub ()
{
    int      func_no ;

    tgr_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,tgr_var,tgr_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS == 0)
    {
        tgr_buff () ;
        scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
        tgr_getsub () ;
        tgr_name () ;
    }
    do
    {
        switch (func_no = tgr_menu())
        {
            case F1:
                tgr_prev () ;
                break ;
            case F2:
                tgr_next () ;
                break ;
        }
    } while (func_no != F9) ;
}

tgr_scr ()
{
    clrscr() ;
    frame (3,5,15,75,"การบันทึกการเกิดเฉพาะรายวิชา") ;
    printp (5,17,"ชื่อ วิชา :",NM_VDO) ;
    printp (6,12,"อาจารย์ผู้สอน :",NM_VDO) ;
    lineH (5,75,7) ;
    printp (9,20,"ชื่อ ภา :",NM_VDO) ;
    printp (10,16,"ภาควิชา :",NM_VDO);
}

tgr_buff ()
{
    strcpy (tgr_buf.tgr_sub,tgr_rec.tgr_sub) ;
    strcpy (tgr_buf.code,tgr_rec.code) ;
    strcpy (tgr_buf.gradel,tgr_rec.gradel) ;
    strcpy (tgr_buf.grade2,tgr_rec.grade2) ;
}

tgr_recc ()
{
    strcpy (tgr_rec.tgr_sub,tgr_buf.tgr_sub) ;
    strcpy (tgr_rec.code,tgr_buf.code) ;

```

```

strcpy (tgr_rec.grade1,tgr_buf.grade1) ;
strcpy (tgr_rec.grade2,tgr_buf.grade2) ;
}

tgr_name ()
{
    int    i;
    char   part2[41];

    strcpy (stu_key0.id,tgr_buf.code) ;
    STATUS = BTRV (B_GETEQ,stu_blk,&stu_rec,&stu_len,&stu_key0,0) ;
    if (STATUS!=0)
    {
        bt_error () ;
        return (FALSE) ;
    }
    printp ( 9,25,"",NM_VDO) ;

    printp ( 9,25,stu_rec.th_name,NM_HL_VDO) ;
    i = 0 ;
    while (stu_rec.th_name[i] != ' ')
        i++ ;
    printp ( 9,27+i,stu_rec.th_sname,NM_HL_VDO) ;

    stu_dep(stu_buf.part,part2);
    printp (10,25,"",NM_VDO);
    printp (10,25,part2,NM_HL_VDO);
    return (TRUE) ;
}

tgr_getsub ()
{
    strcpy (osub_key0.osub_id,tgr_buf.tgr_sub) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS!=0)
    {
        bt_error();
        b_error ("ไม่มีการเปิดสอนวิชานี้") ;
        return (FALSE) ;
    }
    printp (6,25,"",NM_VDO);
    printp (6,25,osub_rec.osub_tea,NM_HL_VDO);
    strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS != 0)
        return (FALSE) ;
    printp (5,25,"",NM_VDO);
    printp (5,25,subj_rec.subj_name,NM_HL_VDO) ;
    return (TRUE) ;
}

```

```
                {23,20,"2:พล.4",RV_VDO},
                {23,53,"9:Exit",RV_VDO}
            } ;

            int      i ;

            i = get_func (F2,fn_tgr) ;
            return (i) ;
        }

tgr_prev ()
{
    STATUS = BTRV (B_GETPR,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("การประมวลผลแฟ้มแล้วเสร็จ") ;
        return ;
    }
    tgr_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub () ;
    tgr_name () ;
}

tgr_next ()
{
    STATUS = BTRV (B_GETNX,tgr_blk,&tgr_rec,&tgr_len,&tgr_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("การประมวลผลแฟ้มแล้วเสร็จ") ;
        return ;
    }
    tgr_buff () ;
    scr_mgr (OLD,DISP_FLAG,OLD,tgr_var,tgr_lbl,4,NULL,NULL) ;
    tgr_getsub () ;
    tgr_name () ;
}
```

```

#include "extern.inc"
#include "stue.inc"

gen_subject ()
{
    int    func_no ;

    gens_scr () ;
    scr_mgr (NEW,DISP_FLAG,NEW,gens_var,gens_lbl,4,NULL,NULL) ;
    STATUS = BTRV (B_GETLW,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;
    if (STATUS ==0)
    {
        gens_rec2buf () ;
        scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
        gens_print () ;
    }
    gens_clr0 () ;
    do
    {
        switch (func_no= gens_menu())
        {
            case F1 :
                gens_prev () ;
                break;
            case F2 :
                gens_next () ;
                break;
        }
    } while (func_no != F9) ;
}

gens_scr ()
{
    clrscr () ;
    frame (3,1,12,79,"หมวดข่าวต่างประเทศ") ;
    lineH (1,79,6) ;
    printp (8,12,"ข่าวเช้า : ",NM_VDO) ;
    printp (9,11,"หน่วยข่าว : ",NM_VDO) ;
    printp (10,13,"ทฤษฎี : ",NM_VDO) ;
    printp (10,27,"โปรแกรม : ",NM_VDO) ;
}

gens_rec2buf ()
{
    gens_buf.gens_dep = gens_rec.gens_dep ;
    gens_buf.gens_year = gens_rec.gens_year ;
    gens_buf.gens_sem = gens_rec.gens_sem ;
    strcpy (gens_buf.code,gens_rec.code) ;
}

gens_buf2rec ()
{
    gens_rec.gens_dep = gens_buf.gens_dep ;
    gens_rec.gens_year = gens_buf.gens_year ;

```



```

gens_rec.gens_sem = gens_buf.gens_sem ;
strcpy (gens_rec.code,gens_buf.code) ;
}

gens_print ()
{
    gens_dep ();
    gens_sem ();
    gens_getsub ();
}

gens_dep ()
{
    static char *dep[] = {"คณิตศาสตร์และวิทยาการคอมพิวเตอร์",
        "เคมีอุตสาหกรรม",
        "เทคโนโลยีชีวภาพ",
        "ฟิสิกส์ประยุกต์",
        "สังคมประยุกต์"} ;

    char dbuf[41];

    strcpy (dbuf,dep[gens_buf.gens_dep]) ;
    printp (4,20,"",NM_VDO) ;
    printp (4,20,dbuf,NM_HL_VDO) ;
}

gens_sem ()
{
    static char *sem[] = {"1st",
        "2nd",
        "Summer" } ;

    char sbuf[7] ;

    strcpy (sbuf,sem[gens_buf.gens_sem]) ;
    printp (5,41,"",NM_VDO) ;
    printp (5,41,sbuf,NM_HL_VDO) ;
}

gens_getsub ()
{
    char cbuf[2], tbuf[2], obuf[2] ;
    char gbuf[61] ;
    int n ;

    strcpy (osub_key0.osub_id,gens_buf.code) ;
    STATUS = BTRV (B_GETEQ,osub_blk,&osub_rec,&osub_len,&osub_key0,0) ;
    if (STATUS != 0)
    {
        b_error("ไม่พบการเปิดสอนกระบวนวิชา") ;
        return (FALSE) ;
    }
    strcpy (subj_key0.subj_code,osub_rec.osub_id) ;
    STATUS = BTRV (B_GETEQ,subj_blk,&subj_rec,&subj_len,&subj_key0,0) ;
    if (STATUS != 0)
    {
        b_error("ไม่พบกระบวนวิชา") ;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามเผยแพร่เอกสารฉบับนี้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        return (FALSE) ;
    }
    strcpy (gbuf,subj_rec.subj_name) ;
    n = 0;
    while (gbuf[n] != ' ' || gbuf[n+1] != ' ')
        n++ ;
    gbuf[n] = '\0' ;
    printp (8,20,"
NM_VDO) ;
    printp (8,20,gbuf,NM_HL_VDO) ;
    itoa_1 (subj_rec.subj_cred,cbuf,1) ;
    printp (9,20,cbuf,NM_HL_VDO) ;
    itoa_1 (subj_rec.subj_theory,tbuf,1) ;
    printp (10,20,tbuf,NM_HL_VDO) ;
    itoa_1 (subj_rec.subj_opr,obuf,1) ;
    printp (10,34,obuf,NM_HL_VDO) ;
    return (TRUE) ;
}

gens_clr0 ()
{
    if (gens_buf.code[0] == ' ')
    {
        printp (8,20,"
NM_VDO) ;
        clr_win (9,20,10,22) ;
        clr_win (10,34,10,50) ;
    }
}

gens_menu ()
{
    static struct lbl_template fn_gens[] = {{23, 4,"1:ก.อน",RV_VDO},
                                             {23,20,"2:ทล.๑",RV_VDO},
                                             {23,53,"9:Exit",RV_VDO}
                                             } ;

    int    i ;

    i = get_func (F2,fn_gens) ;
    return (i) ;
}

gens_prev ()
{
    STATUS = BTRV (B_GETPR,gens_blk,&gens_rec,&gens_len,&gens_key0,0) ;
    if (STATUS != 0)
    {
        b_error ("กด๑ขอชมจอแรกแล้วครับ") ;
        return ;
    }
    gens_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
    gens_print () ;
    gens_clr0 () ;

```

```
gens_next ()
{
    STATUS = BTRV (B_GETNX,gens_blk,&gens_rec,&gens_len,&gens_key0;0) ;
    if (STATUS != 0)
    {
        b_error ("การเชื่อมโยงข้อมูลผิดพลาด") ;
        return ;
    }
    gens_rec2buf () ;
    scr_mgr (OLD,DISP_FLAG,OLD,gens_var,gens_lbl,4,NULL,NULL) ;
    gens_print () ;
    gens_clr0 () ;
}
```



เอกสารอ้างอิง

Borland International Turbo C Reference Manual Borland International,
1989

A Novell Computer Btrieve Record Manager SoftCraft, Inc., 1987

William L. Schweber, Data Communication, McGraw-Hill International
Editions, Electrical & Electronic Engineering Series, 1988.

Herbert Schidt, C Power User's Guide, Osborne McGraw-Hill
International, 1987

Brendan Tangney and Donal O'Mahony, Local Area Networks and their
Applications, Prentice Hall International (UK) Ltd., Hemmel
Hempstead, Hertfordshire, First published 1988.

Sheldon Tom, Novell Netware : The Complete Reference, McGraw hill
Inc., USA., 1990.