

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การสร้างโปรแกรมคอมพิวเตอร์สำหรับ
การบรรจุกล่องในตู้คอนเทนเนอร์



นายคณาพจน์ สวงนวิทย์
นางสาวนัตพร นุชแผน
นายศิริชัย พุ่มจันทร์

เลขหมู่.....
เลขทะเบียน...62759
วัน,เดือน,ปี...21 ส.ค. 2549

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมอุตสาหการ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**CREATING COMPUTER PROGRAM
FOR BIN PACKING PROBLEM**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INDUSTRIAL ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2005**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญาโท

หัวข้อปริญญาโท

การสร้างโปรแกรมคอมพิวเตอร์สำหรับการบรรจุกล่องในตู้คอนเทนเนอร์
CREATING COMPUTER PROGRAM FOR
BIN PACKING PROBLEM

นักศึกษา

นายคณาพงษ์ สงวนรัตน์ รหัสประจำตัว 46015700
นางสาวนันทพร นุชแผน รหัสประจำตัว 46015713
นาย ศิริชัย ทุมจันทร์ รหัสประจำตัว 46015731

หลักสูตร

วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอุตสาหกรรม

อาจารย์ผู้ควบคุมปริญญาโท

(ผศ.ดร.ธรรพสิทธิ์ ถิ่นนรินทร์)

(อาจารย์เชาวลิต หามนตรี)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์
นักศึกษา

การสร้างโปรแกรมคอมพิวเตอร์สำหรับการบรรจุกล่องในตู้คอนเทนเนอร์
นายคณาพจน์ สวงวรักษ์
นางสาวนัตพร นุชแผน
นายศิริชัย พุ่มจันทร์

หลักสูตร

วิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมอุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา

2548

อาจารย์ผู้ควบคุมปริญญานิพนธ์

ผศ.ดร. สรรพสิทธิ ถิ่นนรรค์
อาจารย์เชาวลิต หามนตรี

บทคัดย่อ

ปัญหาการจัดวางสินค้าลงบนตู้คอนเทนเนอร์ เป็นปัญหาที่มีความสำคัญต่อระบบการขนส่งในปัจจุบัน เพราะจะมีส่วนอย่างมากในการช่วยลดต้นทุนการขนส่ง โดยปัญหาการจัดวางสินค้าคือปัญหาที่จะต้องกำหนดจำนวนกล่องจัดเรียงเข้าไปในตู้คอนเทนเนอร์ ให้เหลือเนื้อที่น้อยที่สุด และจัดเรียงกล่องจำนวนสินค้าให้ได้มากที่สุด โดยมีการคิดในแบบ 3 มิติ ปริญญานิพนธ์ฉบับนี้เป็นการเสนอสร้างโปรแกรมคอมพิวเตอร์สำหรับค้นหาคำตอบที่เหมาะสม ในปัญหาการจัดวางสินค้าในตู้คอนเทนเนอร์ด้วยวิธีวิวิธคณิต ผลการทดลองได้แสดงให้เห็นว่าโปรแกรมการจัดเรียงสินค้าในตู้คอนเทนเนอร์สามารถจัดเรียงสินค้าและให้ค่าเฉลี่ยของคำตอบที่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title **Creating Computer Program for Bin Packing Problem**

Student **Mr.Kanapoj Sa-nguanrak**
Miss Nattaporn Nuchpan
Mr.Sirichai Pumjan

Degree **Bachelor of Engineering in Industrial Engineering**
King Mongkut's institute of Technology Ladkrabang

Academic Year **2005**

Thesis Advisor **Asist.Pro.Dr. Sunpasit Limuararat**
Mr. Chouwalit Hamontree

ABSTRACT

The bin – packing problem is the problem of orthogonally packing a set of boxes into a minimum number of three – dimensional bins. In this project to present a heuristic algorithm based on guided local search. The presented algorithm iteratively decreases the number of bids, each time searching for a feasible packing of the boxes. The algorithm can also be applied to two – dimensional bin – packing problem by having a constant depth all boxes and bins. Computational experiments are reported the arrange the boxes into the container showing that the algorithm on average finds better solutions.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญาโทสำเร็จขึ้นมาได้จากความช่วยเหลือจาก อาจารย์ทุกท่านที่คณะวิศวกรรมอุตสาหกรรม ที่ได้เป็น คณะกรรมการที่ปรึกษา ที่ได้ให้ข้อเสนอแนะเพิ่มเติมในการทำปริญญาโทนี้ ทำให้การทำปริญญาโทนี้ได้สำเร็จ และบรรลุตามวัตถุประสงค์

ขอขอบคุณ ศส.ดร.สรรพลักษณ์ สีมันรัตน์ และ อาจารย์เชาวลิต หามนตรี ที่ปรึกษาปริญญาโทนี้ ซึ่งได้ กรุณาให้คำปรึกษา แนะนำเอกสาร วารสาร และหนังสือที่มีประโยชน์ต่อการทำปริญญาโทนี้ ตลอดจนช่วยแก้ไข ปริญญาโทนี้ และช่วยแนะนำในการจัดทำปริญญาโทนี้ และเป็นผู้ทดสอบในการใช้งานของโปรแกรม และ ตรวจสอบข้อผิดพลาดของโปรแกรมเพื่อนำกลับมาแก้ไขให้สมบูรณ์

สุดท้ายนี้กลุ่มข้าพเจ้าขอขอบพระคุณ บิคา มารดา ที่ได้ให้ความสนับสนุนโดยตลอด ขอขอบพระคุณ อาจารย์ ทุกท่าน และสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้อบรมสั่งสอน ให้ความรู้ ให้จริยธรรมใน ด้านต่างๆ และฝึกทักษะการทำงาน เพื่อให้สามารถนำไปใช้ให้เกิดประโยชน์ในชีวิตประจำวันต่อสังคมและ ประเทศชาติในอนาคต

นายคณาพนธ์ สวงนรักษ์
นางสาวนัตพร นุชแสน
นายศิริชัย พุ่มจันทร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	ก
บทคัดย่อภาษาอังกฤษ.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช
บทที่ 1 บทนำ	
1.1 ความสำคัญของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบเขตของโครงการ.....	1
1.4 ขั้นตอนการดำเนินงาน.....	1
1.5 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	
2.1 วิธีทางทฤษฎี.....	3
2.2 วิธีสถิติ.....	4
2.3 วิธีเชิงรวม.....	7
บทที่ 3 วิธีการดำเนินงาน	
3.1 ศึกษาและรวบรวมข้อมูล.....	8
3.2 วิธีการดำเนินงาน.....	8
3.2.1 ออกแบบอัลกอริทึมของสถิติ.....	8
3.2.2 กำหนดปัจจัยต่างๆที่นำมาพิจารณาในการแก้ปัญหาทางด้านการบรรจุ.....	8
3.3 โครงสร้างในการสร้างโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	9
3.3.1 ส่วนรับข้อมูลของโปรแกรม จะประกอบด้วย.....	9
3.3.2 ส่วนการทำงานของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	10
3.3.3 ส่วนการแสดงผลลัพธ์ของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	11
3.4 องค์ประกอบของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	13
3.4.1 ส่วนการรับข้อมูลของโปรแกรม.....	13
3.4.2 ส่วนการแสดงผลลัพธ์ของโปรแกรม.....	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการดำเนินงาน	
4.1 ทดสอบโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	16
4.2 การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (X, Y, Z).....	19
4.3 ประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์.....	26
4.4 ตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ.....	27
บทที่ 5 สรุปผลการดำเนินงาน	
5.1 การสรุปผลการดำเนินงาน.....	29
5.2 ข้อเสนอแนะ.....	29
หนังสืออ้างอิง.....	31
ภาคผนวก ก.....	ผก 1
ภาคผนวก ข.....	ผข 1
ภาคผนวก ค.....	ผค 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 4.1 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 1	17
ตารางที่ 4.2 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 2	17
ตารางที่ 4.3 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 3	18
ตารางที่ 4.4 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 4	18
ตารางที่ 4.5 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (x, y, z) ของกรณีศึกษาข้อมูลชุดที่ 1 ...	19
ตารางที่ 4.6 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (x, y, z) ของกรณีศึกษาข้อมูลชุดที่ 2 ...	21
ตารางที่ 4.7 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (x, y, z) ของกรณีศึกษาข้อมูลชุดที่ 3 ...	23
ตารางที่ 4.8 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (x, y, z) ของกรณีศึกษาข้อมูลชุดที่ 4 ...	25
ตารางที่ 4.9 แสดงประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์	26
ตารางที่ ผค 1 ข้อมูลคอนเทนเนอร์ขนาดต่างๆ	ผค 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 3.1 รูปแสดงการหมุนของกล่องบรรจุภัณฑ์.....	9
รูปที่ 3.2 Flow Chart แสดงขั้นตอนการทำงานของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์.....	10
รูปที่ 3.3 แสดงตำแหน่งพิกัดที่ (0, 0, 0) ของตู้คอนเทนเนอร์ และบรรจุภัณฑ์.....	12
รูปที่ 3.4 แสดงจุดพิกัดที่สามารถทำการวางบรรจุภัณฑ์ขึ้นไป.....	12
รูปที่ 3.5 แสดงส่วนการรับข้อมูลของโปรแกรม.....	13
รูปที่ 3.6 แสดงตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (X, Y, Z).....	14
รูปที่ 3.7 แสดงประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์.....	14
รูปที่ 3.8 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ.....	15
รูปที่ 4.1 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 1.....	27
รูปที่ 4.2 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 2.....	27
รูปที่ 4.3 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 3.....	28
รูปที่ 4.4 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 4.....	28
รูปที่ ผก 1 จอหลักของ โปรแกรม.....	ผก 1
รูปที่ ผก 2 แสดงหน้าจอในส่วนการรับข้อมูลของ โปรแกรม.....	ผก 2
รูปที่ ผก 3 แสดงหน้าจอในส่วนการรับข้อมูลของ โปรแกรม.....	ผก 3
รูปที่ ผก 4 แสดงหน้าจอในส่วนการรับข้อมูลของ โปรแกรม.....	ผก 3
รูปที่ ผก 5 แสดงหน้าจอในส่วนการรับข้อมูลของ โปรแกรม.....	ผก 4
รูปที่ ผก 6 แสดงหน้าจอบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต.....	ผก 4
รูปที่ ผก 7 แสดงหน้าจอบอกประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์.....	ผก 5
รูปที่ ผก 8 แสดงหน้าจอบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ.....	ผก 5
รูปที่ ผก 9 แสดงการเลือกคำสั่ง Show Animated.....	ผก 6
รูปที่ ผก 10 แสดงการเลือกคำสั่งออกจากโปรแกรม.....	ผก 6

บทที่ 1

บทนำ

1.1 ความสำคัญของโครงการ

ปัญหาการออกแบบการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ ในธุรกิจต่างๆมีการออกแบบการจัดวางบรรจุภัณฑ์หลากหลายวิธี เนื่องจากปัญหาการออกแบบการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์เป็นปัญหาที่มีความซับซ้อนสูงมาก ดังนั้นโปรแกรมนี้จะช่วยในการทำให้เกิดการจัดวางกล่องบรรจุภัณฑ์ให้เกิดพื้นที่สูญเปล่าน้อยที่สุด เพื่อช่วยในการลดต้นทุนการขนส่งหรือขนย้าย เช่น ค่าเช่าตู้คอนเทนเนอร์ ค่าเช่ารถหิ้วลากตู้คอนเทนเนอร์ และในบางครั้งอาจจะลดความผิดพลาดในการออกแบบซึ่งส่งผลให้เกิดความเสียหายในด้านการค้า เช่น เสียค่าปรับในการส่งสินค้าล่าช้า ดังนั้นปัญหาการออกแบบการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ ควรมีการใช้ความรู้ทางวิชาการมาประยุกต์และพัฒนา เพื่อให้เกิดประสิทธิภาพสูงในการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ และยังสามารถป้องกันความผิดพลาดในการออกแบบการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อหาวิธีในการจัดวางบรรจุภัณฑ์ เพื่อให้มีปริมาตรสูญเปล่าน้อยที่สุด
2. ศึกษาและออกแบบโปรแกรมคอมพิวเตอร์ เพื่อนำมาเป็นเครื่องมือช่วยในการตัดสินใจการจัดวางบรรจุภัณฑ์ในรูปแบบ 3 มิติได้อย่างเหมาะสม
3. สร้างโปรแกรมคอมพิวเตอร์ที่ใช้ในการออกแบบการจัดวางบรรจุภัณฑ์

1.3 ขอบเขตของโครงการ

1. ลักษณะรูปทรงของตัวบรรจุภัณฑ์และพื้นที่จัดเก็บต้องมีรูปทรงสี่เหลี่ยมเท่านั้น
2. ขนาดของบรรจุภัณฑ์สามารถมีความกว้าง ยาว และสูงได้แตกต่างกันอย่างเป็นอิสระ
3. บรรจุภัณฑ์สามารถวางได้ด้านเดียว แต่สามารถหมุนได้ตามแนวราบเท่านั้น

1.4 ขั้นตอนการดำเนินงาน

การวางแผนการดำเนินงานได้แบ่งขั้นตอนออกเป็น 6 ขั้นตอนดังต่อไปนี้

1. ศึกษาและเก็บรวบรวมข้อมูล
2. ศึกษาการใช้งานของโปรแกรมวิซิวัล เบสิก
3. ออกแบบส่วนต่าง ๆ ของโปรแกรม
4. เขียนโปรแกรม
5. ทำการทดลองการทำงานและแก้ไขส่วนบกพร่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และนำอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. จัดทำรูปเล่มปริญญานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถระบุรูปแบบการจัดวางบรรจุภัณฑ์เข้าสู่พื้นที่จัดเก็บได้อย่างเหมาะสม
2. ช่วยให้เปอร์เซ็นต์การใช้ปริมาตรของพื้นที่จัดเก็บมากที่สุด
3. ช่วยลดเวลาในการออกแบบการจัดวางบรรจุภัณฑ์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

การแก้ปัญหาการออกแบบขนาดของบรรจุภัณฑ์และการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์สามารถแบ่งออกได้เป็น 4 ประเภท ดังนี้

1. วิธีทางทฤษฎี (Theoretical Approach) เป็นวิธีที่สามารถหาผลลัพธ์ที่ดีที่สุด โดยใช้ความรู้ในวิชาการวิจัยการดำเนินงานมาประยุกต์ ได้แก่ โปรแกรมเชิงเส้นตรง โปรแกรมเชิงพลวัต วิธีเบรอันซ์ แอนด์ บาวด์ วิธีในกลุ่มนี้จะเหมาะกับปัญหามาตรฐานใหญ่วิธีนี้จะมีความยุ่งยาก และใช้เวลาคำนวณมาก
2. วิธีฮิวริสติก (Heuristic Approach) เป็นวิธีที่ใช้ชุดของกฎเกณฑ์ที่กำหนดขึ้นช่วยในการหาผลลัพธ์ผลลัพธ์ที่ได้จากวิธีนี้ไม่สามารถพิสูจน์ได้ว่าเป็นผลลัพธ์ที่ดีที่สุด แต่มักจะให้ผลลัพธ์ที่น่าพอใจและใกล้เคียงกับผลลัพธ์ที่ดีที่สุด ข้อดีของวิธีนี้คือสามารถแก้ปัญหาขนาดใหญ่ได้โดยใช้เวลาในการคำนวณไม่มากนัก
3. วิธีเชิงรวม (Hybrid Approach) เป็นวิธีที่ใช้หลักเกณฑ์จาก 3 วิธีแรกประกอบกันเพื่อหาคำตอบ ข้อดีของวิธีในกลุ่มนี้คือการรวมข้อดีของแต่ละวิธีมาใช้ในการแก้ปัญหา

2.1 วิธีทางทฤษฎี (Theoretical Approach)

วิธีนี้อาจใช้แบบจำลองทางคณิตศาสตร์มาช่วยในการแก้ปัญหา หรืออาจใช้วิธีตรวจสอบรูปแบบผลลัพธ์ที่เป็นไปได้ทุกรูปแบบ ทำให้ผลลัพธ์ที่ได้เป็นผลลัพธ์ที่ดีที่สุด วิธีในกลุ่มนี้เหมาะกับปัญหาที่มีขนาดไม่ใหญ่มากนัก เนื่องจากรูปแบบการจัดวางที่เป็นไปได้จะเพิ่มแบบอีกไปเนิ่นเนิ่นกับจำนวนวัตถุ

Scheithauer Terno (1993) ได้เสนอโมเดลในการแก้ปัญหาการบรรจุวัตถุที่มีขนาดในหนึ่งมิติ และสองมิติ ทั้งที่เป็น คอนเว็กซ์ (Convex) และนอนคอนเว็กซ์ โพลีกอน (Non-Convex Polygon) โดยใช้โปรแกรมแบบเชิงเส้น (Linear Optimization) ร่วมกับตัวแปร 0,1

Scheithauer (1995) ได้ทำการศึกษาการแก้ปัญหาการบรรจุ (Packing Problem) โดยวิธีเบรอันซ์ แอนด์ บาวด์ พบว่าสามารถลดเวลาในการคำนวณลงได้ ด้วยการตรวจสอบรูปแบบการเรียงว่ามีลักษณะคล้ายคลึงกันหรือไม่ ถ้าพบว่ารูปแบบการเรียงคล้ายคลึงกัน จะใช้ผลลัพธ์ที่จัดเก็บไว้ในหน่วยความจำแทนการคำนวณ ทำให้ลดเวลาคำนวณในขั้นตอนย่อย ๆ ลงได้

Fekete and Schepers (1997a) ทำการศึกษาปัญหาการบรรจุวัตถุ ที่มีมิติมากกว่าหนึ่งมิติ โดยใช้ทฤษฎีกราฟ (Graph-Theoretical) เพื่อระบุขอบเขตของคำตอบที่เป็นไปได้ และเสนอวิธีการหาค่าขอบเขตล่าง (Lower Bound) คำตอบในปัญหาการบรรจุ การได้ค่าขอบเขตล่างที่ดีทำให้การหาคำตอบด้วยวิธี เบรอันซ์ แอนด์ บาวด์ (Branch and Bound Method) ทำได้เร็วขึ้น (Fekete and Schepers, 1997b) นอกจากนี้ยังได้เสนออัลกอริทึมในการแก้ปัญหาการบรรจุด้วยวิธี เบรอันซ์ แอนด์ บาวด์ ร่วมกับทฤษฎีกราฟ และวิธีการหาค่าขอบเขตล่าง (Lower Bound) ในการหาคำตอบ (Fekete and Schepers, 1997c)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 วิธีฮิวริสติก (Heuristic Approach)

เป็นวิธีแก้ปัญหาคำตอบที่ได้รับความนิยมมากที่สุดในปัจจุบัน มีวิธีฮิวริสติกที่คิดค้นขึ้นมาหลายวิธี ได้แก่

George and Robinson (1980) ได้นำเสนอวิธีการหาคำตอบแบบประมาณของปัญหาคำตอบที่จัดวางบรรจุภัณฑ์ในตู้คอนเทนเนอร์ ลักษณะเฉพาะของปัญหาในการจัดวางคือปริมาตรรวมของบรรจุภัณฑ์ทั้งหมด มีปริมาตรน้อยกว่าปริมาตรของตู้คอนเทนเนอร์เล็กน้อย เป้าหมายของการบรรจุคือจัดเรียงบรรจุภัณฑ์ทั้งหมดโดยใช้จำนวนตู้คอนเทนเนอร์น้อยที่สุด วิธีการที่นำเสนอในงานวิจัยนับได้ว่าเป็นวิธีการหาคำตอบแบบประมาณสำหรับปัญหาคำตอบที่จัดวางบรรจุภัณฑ์ในตู้คอนเทนเนอร์แบบแรกที่ถูกเสนอในวารสารในระดับนานาชาติ และมักถูกนำมาใช้เป็นวิธีเปรียบเทียบประสิทธิภาพในการบรรจุในรายงานชิ้นอื่นๆ

Scheithauer (1991) ได้ทำการศึกษาปัญหาการบรรจุวัตถุรูปกล่องลงในภาชนะบรรจุรูปกล่อง โดยเสนออัลกอริทึมในการหาคำตอบแบบประมาณ (Approximation Algorithm) ที่ใช้วิธี "Forward Stat Strategy" ของกำหนดพลวัต (Dynamic Programming) และ "อัลกอริทึม CL" (CL Algorithm) สำหรับการแก้ปัญหาขนาดเล็ก

Bischoff et al. (1995) ได้ทำการศึกษาปัญหาการจัดเรียงบรรจุภัณฑ์บนแพallet (Pallet Loading Problem) เป้าหมายของการบรรจุคือจัดเรียงบรรจุภัณฑ์ให้บรรลุประโยชน์ของการใช้ปริมาตรมีค่ามากที่สุด โดยทราบค่าขนาดของบรรจุภัณฑ์ขนาดของแพallet และขีดจำกัดความสูงของบรรจุภัณฑ์ที่สามารถวางบนแพallet ในงานวิจัยได้นำเสนออัลกอริทึมการหาคำตอบแบบประมาณ (Heuristic Algorithm) ซึ่งมีหลักการดังนี้

1. บรรจุภัณฑ์จะถูกจัดเรียงในแนวราบทีละชั้น
2. พื้นที่ซึ่งมีขนาดเล็กที่สุด จะถูกใช้จัดวางบรรจุภัณฑ์ก่อนพื้นที่ขนาดใหญ่
3. ใช้วิธีการการจัดวางแบบ 2-Blocks รูปแบบการวางที่ให้ค่าอัตราประโยชน์การใช้พื้นที่สูงสุดในแต่ละรอบของการคำนวณ จะถูกเลือกเป็นผังการจัดวาง
4. ย้อนกลับไปยังขั้นตอนที่ 2 จนบรรจุภัณฑ์ทั้งหมดถูกจัดวาง

Bischoff and Ratchliff (1995) ทำการศึกษา ปัญหาการจัดเรียงบรรจุภัณฑ์บนแพallet โดยจำนวนแพallet ที่ต้องใช้มีจำนวนมากกว่าหนึ่งแพallet เป้าหมายของการบรรจุคือต้องการให้จำนวนแพallet ที่ต้องใช้มีจำนวนน้อยที่สุด เมื่อทราบค่าขนาดของบรรจุภัณฑ์ ขนาดของแพallet และขีดจำกัดความสูงของบรรจุภัณฑ์ที่สามารถวางบนแพallet ในการวิจัยได้ประยุกต์หลักการจัดเรียงแบบ "Single-Pallet Algorithm" จากงานวิจัยของ Bischoff et al. (1995) ให้สามารถจัดเรียงบรรจุภัณฑ์บนแพallet จำนวนมากกว่าหนึ่งได้ นอกจากนี้ในงานวิจัยได้เสนออัลกอริทึมสำหรับการจัดบรรจุภัณฑ์ลงบนหลายแพallet โดยเฉพาะ เรียกว่า "Simultaneous Approach"

ผลจากการเปรียบเทียบอัตราประโยชน์การใช้ปริมาตรระหว่าง "Single-Pallet Algorithm" กับ "Simultaneous Approach" พบว่า "Single-Pallet Algorithm" ให้ค่าผลลัพธ์ที่ดีกว่า ดังนั้นจึงมีการเสนอวิธีปรับปรุง "Simultaneous Approach" โดยกำหนดความสำคัญ (Priority) ให้กับกล่องตามขนาดของกล่อง (ด้วยวิธีการจัดความสำคัญ 2 รูปแบบ) ผลจากการปรับปรุงพบว่าคำตอบที่ได้ มีผลที่ดีกว่าวิธีการ "Simultaneous Approach" แบบเดิม

Scheithauer and Terno (1996) ได้เสนอวิธีการหาค่าตอบแบบประมาณ (Heuristic for Two-Dimensional orthogonal) ที่มีชื่อว่า "ฮิวริสติก G4" ในการแก้ปัญหาการจัดเรียงบรรจุภัณฑ์บนแพลตฟอร์ม (Pallet Loading Problem) โครงสร้างของ G4 จะใช้หลักการบรรจุกล่องลงในบล็อกโดยมีการกำหนดตำแหน่งเหมือนกัน จากการทดสอบการจัดเรียงกล่องที่มีปริมาณมากกว่า 50 กล่องพบว่า "ฮิวริสติก G4" ให้ผลลัพธ์ที่ประมาณ 99% ของค่าที่ดีที่สุด และแม้ว่าอัลกอริทึมจะมีลักษณะเป็นแบบกึ่งโพลิโนเมียล (Pseudo-Polynomial) แต่ผลการทดสอบกับจัดเรียงกล่องที่มีปริมาณมากกว่า 200 กล่อง พบว่า "ฮิวริสติก G4" ใช้เวลาในการคำนวณไม่มากนัก

Scheithauer and Sommerwei (1996) ได้ศึกษาปัญหาการบรรจุรูปสี่เหลี่ยมขนาดเล็กลงในสี่เหลี่ยมขนาดใหญ่ (Rectangle Packing Problem:RPP) เพื่อให้การใช้พื้นที่มีประสิทธิภาพมากที่สุด โดยได้พัฒนาวิธีการหาค่าตอบแบบประมาณ (Heuristic) โดยใช้หลักการจาก "ฮิวริสติก G4" ใช้เวลาในการคำนวณไม่มาก

Scheithauer and Sommerwei (1996) ได้ศึกษาปัญหาการบรรจุรูปสี่เหลี่ยมขนาดเล็กลงในสี่เหลี่ยมขนาดใหญ่ (Rectangle Packing Problem:RPP) เพื่อให้การใช้พื้นที่มีประสิทธิภาพมากที่สุด โดยได้พัฒนาวิธีการหาค่าตอบแบบประมาณ (Heuristic) โดยใช้หลักการจาก "ฮิวริสติก G4" เพื่อหาค่าตอบของปัญหาการจัดเรียงบรรจุภัณฑ์บนแพลตฟอร์ม (Pallet Loading Problem)

Gradisar et al.(1997) ทำการศึกษา ปัญหาเศษผ้าที่เหลือจากการตัดม้วนในอุตสาหกรรมเสื้อผ้า ซึ่งจากการสำรวจพบว่าในอุตสาหกรรมประเภทดังกล่าวจะมีเศษผ้าเหลือซึ่งไม่สามารถนำมาใช้ประโยชน์ได้ (Waste) จากการตัดประมาณ ร้อยละ 2 ถึงร้อยละ 3 ในการทำงานแบบชิ้นส่วนของเสื้อผ้า (Pattern Parts) จะถูกจัดวางลงในผ้าที่มีรูปร่างสี่เหลี่ยม (Pattern Shape) ลงบนผ้าม้วนให้มีเศษผ้าเหลือน้อยที่สุด สำหรับในการแก้ปัญหาการตัดม้วนได้เคยมีผู้จัดทำมาก่อนแล้ว แต่สิ่งที่ทำให้อัลกอริทึมในการแก้ปัญหาของการวิจัยครั้งนี้แตกต่างจากการวิจัยก่อนหน้านี้คือสามารถแก้ปัญหาการตัด เมื่อม้วนผ้าวัตถุดิบที่ใช้มีความยาวไม่เท่ากัน

Liu and Hsiao (1997) ทำการศึกษา ปัญหาการจัดเรียงบรรจุภัณฑ์บนแพลตฟอร์มทั้งหมดมีขนาดเท่ากัน โดยให้มีวัตถุประสงค์ในการจัดเรียง 2 ประการ เรียงตามลำดับความสำคัญคือ

1. มุ่งอรรถประโยชน์ของการใช้ปริมาตรสูงสุด (Cube Utilization) หมายถึง

$$\text{MAX} \left[\frac{\sum \text{ปริมาตรบรรจุภัณฑ์} \times 100}{\text{พื้นที่ของแพลตฟอร์ม} \times \text{ซิดจังก์กิตความสูง}} \right]$$

2. มุ่งเสถียรภาพของบรรจุภัณฑ์ เนื่องจากการบรรจุภัณฑ์บนแพลตฟอร์มซึ่งต้องถูกขนย้ายโดยไม่มีกรรไกรค้ำยันด้านข้าง ดังนั้นปัจจัยเสถียรภาพจึงเป็นปัจจัยที่ให้ความสำคัญ

ในการวิจัยได้เสนอวิธีการหาค่าตอบซึ่งประกอบด้วย 5 ขั้นตอน คือ

1. หารูปแบบการจัดเรียงที่เป็นไปได้ และคำนวณจำนวนชิ้นของบรรจุภัณฑ์ต่อชิ้น
2. เพื่อให้การจัดเรียงในขั้นตอนที่ 1 มีความหลากหลายจะทำการสลับตำแหน่งให้การจัดเรียงบรรจุภัณฑ์ในแนวระนาบแตกต่างกัน เป้าหมายของขั้นตอนนี้คือเพื่อให้การจัดเรียงระหว่างชั้นบรรจุภัณฑ์ จะไม่เกิดสภาพเรียงต่อกันเป็นแท่ง ซึ่งทำให้เสถียรภาพลดลง
3. หาจำนวนกล่องที่สามารถจัดเรียงได้ในแพลตฟอร์ม (มีซิดจังก์กิตความสูง)
4. คำนวณค่าสัมประสิทธิ์เสถียรภาพของการจัดเรียงระหว่างชั้นต่อชิ้นในทุกกรณีที่เป็นไปได้
5. หาวิธีการเรียงที่ให้ค่าเสถียรภาพรวมสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Terno et al.(1997) ทำการศึกษา ปัญหาการจัดเรียงบรรจุภัณฑ์บนแพลงเกต โดยให้แพลงเกตที่ใช้การวางบรรจุภัณฑ์น้อยที่สุด วิธีการแก้ปัญหาทำโดยการสร้างอัลกอริทึมเพื่อหาวิธีใช้ประโยชน์ของปริมาตรให้สูงสุด ภายใต้ข้อจำกัดในการปฏิบัติ (ข้อจำกัดทางเทคโนโลยี, ข้อจำกัดของน้ำหนักบรรจุภัณฑ์ของแพลงเกต, เสถียรภาพของบรรจุภัณฑ์บนแพลงเกต) แนวคิดที่ใช้คือ วิธี บรรณซ์ แอนด์ บาวด์ และ การหาค่าตอบที่ดีแบบประมาท (Heuristic) สำหรับวิธีแก้ปัญหการจัดเรียงบรรจุภัณฑ์บนแพลงเกตที่พัฒนาขึ้น การหาค่าตอบของการจัดวางในระบบสามมิติจะใช้วิธีการจัดวางบรรจุภัณฑ์ทีละชั้น ซึ่งรูปแบบการหาค่าที่ดีที่สุดของการจัดวางแบบสองมิติในการแก้ปัญหา

Miyazawa and Wakabayashi (1997a) ทำการศึกษาปัญหาของการบรรจุภัณฑ์วัตถุสามมิติซึ่งมีลักษณะดังนี้ 1) ให้รายละเอียดของวัตถุรูปกล่องสี่เหลี่ยมที่จะถูกบรรจุ ได้แก่ ความกว้าง ความยาว และความสูง 2) ให้ขนาดของอุปกรณ์บรรจุซึ่งมีลักษณะเป็นกล่องสี่เหลี่ยม ที่กำหนดขนาดหน้าตัด แต่ความสูงมีขนาดไม่จำกัด วัตถุประสงค์ของการบรรจุ คือต้องการให้ความสูงของอุปกรณ์บรรจุมีค่าต่ำที่สุด ในการวิจัยเสนออัลกอริทึมในการหาค่าตอบแบบประมาท (Approximation Algorithm) เมื่อทดลองกับปัญหาจริง พบว่าค่าเขตจำกัดของประสิทธิภาพ (Asymptotic Performance Bound) มีค่าอยู่ระหว่าง 2.5 ถึง 2.67 ผลลัพธ์ที่ได้สามารถแก้ปัญหาซึ่งตั้งขึ้นโดย K. Li and K-H. Cheng ว่ามีอัลกอริทึมในการหาค่าตอบแบบประมาทแบบใดที่ให้ค่าเขตจำกัดของประสิทธิภาพต่ำกว่า 2.89

Miyazawa and Wakabayashi (1997b) ทำการศึกษาปัญหาของการบรรจุวัตถุสามมิติวัตถุประสงคของการบรรจุที่ต้องการ คือความสูงของอุปกรณ์บรรจุมีค่าต่ำที่สุด ในการวิจัยได้เสนออัลกอริทึม “Orthogonal Z-Oriented Three-Dimensional Packing Problem (TPP)” และวิเคราะห์ค่าเขตจำกัดของประสิทธิภาพ (Asymptotic Performance Bound) ของปัญหาการบรรจุกล่องสี่เหลี่ยม $L=(b_1, b_2, \dots, b_n)$ ลงในกล่องสี่เหลี่ยม $B=(l, w, \infty)$ ในการวิจัยได้แบ่งปัญหาการบรรจุออกเป็น 2 กลุ่ม กลุ่มแรกเป็นปัญหาการบรรจุโดยให้วัตถุที่ถูกบรรจุสามารถหมุนได้ 90° รอบแกน

Z เรียกว่าปัญหาแบบ TPP และกลุ่มที่สอง เป็นปัญหาแบบ TPP' ซึ่งก็คือปัญหาแบบ TPP ซึ่งลดรูปลงมา โดยไม่อนุญาตให้วัตถุที่ถูกบรรจุสามารถหมุนได้ 90° รอบแกน Z อัลกอริทึม (Algorithm) ของปัญหาแบบ TPP' เรียกว่า “อัลกอริทึม R” นอกจากนี้ในรายงานยังได้แสดงอัลกอริทึมสำหรับกรณีเฉพาะได้แก่ “อัลกอริทึม LS” สำหรับวัตถุที่ถูกบรรจุมีหน้าตัดของฐานเป็นสี่เหลี่ยมจัตุรัส “อัลกอริทึม BS” สำหรับอุปกรณ์บรรจุมีหน้าตัดของฐานเป็นสี่เหลี่ยมจัตุรัสและ “อัลกอริทึม SS” สำหรับวัตถุที่ถูกบรรจุและอุปกรณ์บรรจุมีหน้าตัดของฐานเป็นสี่เหลี่ยมจัตุรัสจากการวิจัยพบว่าเมื่อกำหนด A แทน อัลกอริทึมในการหาค่าตอบและ $\rho(A)$ เป็นค่าเขตจำกัดของประสิทธิภาพของอัลกอริทึม ค่าของ $\rho(A)$ สำหรับปัญหาแต่ละประเภทมีค่าอยู่ระหว่าง $2.5 \leq \rho(R) < 2.67, 2.5 \leq \rho(LS) < 2.528, 2.5 \leq \rho(BS) < 2.543, 2.333 \leq \rho(SS) < 2.361$ อัลกอริทึมที่แสดงในรายงานฉบับนี้มีความซับซ้อน (เวลาในการคำนวณ) $O(n \log n)$ ซึ่งเท่ากับอัลกอริทึมในการแก้ปัญหาแบบอื่นที่เคยมีผู้นำเสนอให้ แต่อัลกอริทึมที่พัฒนาขึ้นใหม่นี้มีประสิทธิภาพในการบรรจุดีกว่า

Ferreira et al. (1999) ทำการศึกษาปัญหาการบรรจุวัตถุสองมิติ รูปทรงสี่เหลี่ยมจัตุรัส ลงในอุปกรณ์บรรจุรูปทรงสี่เหลี่ยมจัตุรัส วัตถุประสงค์ของการบรรจุคือ ใช้อุปกรณ์บรรจุจำนวนน้อยที่สุด ในรายงานได้เสนออัลกอริทึมในการหาค่าตอบแบบประมาท (Approximation Algorithm) ซึ่งเวลาที่ใช้ในการแก้ปัญหาเพิ่มขึ้นในแบบโพลีโนเมียล (Big O Time) จากการทดสอบอัลกอริทึมในการแก้ปัญหา พบว่าค่าเขตจำกัดของประสิทธิภาพ (Asymptotic Performance Bound) มีค่าอยู่ 1.988 ซึ่งมีประสิทธิภาพดีกว่า อัลกอริทึมของ Chung, Garey and Johnson ที่ให้ค่าเขตจำกัดของประสิทธิภาพเท่ากับ 2.125

2.3 วิธีเชิงรวม (Hybrid Approach)

Lai et al. (1998) ทำการศึกษา วิธีแก้ปัญหาการจัดวางสินค้าในตู้คอนเทนเนอร์ เมื่อต้องจัดวางสินค้าตามลำดับการจัดส่ง การแก้ปัญหาใช้หลักการของแบบจำลองกราฟ (Graph Model) และวิธีการหาคำตอบที่ดีแบบประมาณ (Heuristic) ขนาดของกราฟในการคำนวณ

การจัดเรียงสินค้ามีหลักเกณฑ์ 3 ประการคือ

1. สินค้าที่มีการส่งเข้าที่สุดจะถูกบรรจุก่อน (วางตอนในของตู้คอนเทนเนอร์)
2. สินค้าที่ถูกให้ลูกค้ารายเดียวกันจะถูกจัดวางชิดกัน
3. มีการใช้อรรถประโยชน์ของปริมาตรให้มากที่สุด

ในการทดสอบแก้ปัญหาโดยใช้หลักการของแบบจำลองกราฟและวิธีการหาคำตอบที่ดีแบบประมาณ กับปัญหาที่สามารถหาคำตอบที่ดีที่สุดได้ (ปัญหาขนาดเล็กและขนาดกลางที่ทราบคำตอบแล้ว) พบว่าทั้งสองวิธีหาคำตอบที่ดีที่สุดเช่นเดียวกัน แต่วิธีการหาคำตอบที่ดีแบบประมาณ ทำให้ขนาดของกราฟและเวลาประมวลผล ลดลงอย่างมาก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงาน

ขั้นตอนการดำเนินงานในการออกแบบระบบฐานข้อมูลและสร้างโปรแกรมคอมพิวเตอร์สำหรับการจัดวางกล่องบรรจุภัณฑ์มีขั้นตอนต่างๆ ดังต่อไปนี้

3.1 ศึกษาและรวบรวมข้อมูล

ขั้นตอนการศึกษาและเก็บรวบรวมข้อมูลเพื่อการทำโครงการ เป็นขั้นตอนในการศึกษาเกี่ยวกับหลักการ ทำงานของตัวอย่างโปรแกรม ลักษณะปัญหาการบรรจุแบบเนบแซค (Knapsack) การแก้ไขปัญหาโดยวิธีฮิวริสติก และผลงานวิจัยที่เกี่ยวข้อง

3.2 วิธีการดำเนินงาน

ในการดำเนินการสร้างโปรแกรมคอมพิวเตอร์สำหรับการจัดวางกล่องบรรจุภัณฑ์ ผู้ดำเนินงานได้แบ่งการ ดำเนินการออกเป็นหัวข้อหลักๆ ดังนี้

3.2.1 ออกแบบอัลกอริทึมของฮิวริสติก

เป็นอัลกอริทึมที่ใช้ในการจัดวางบรรจุภัณฑ์ โดยมุ่งเน้นถึงการใช้พื้นที่ในการจัดวางให้เกิดประสิทธิภาพ มากที่สุด อัลกอริทึมนี้จะเป็นเงื่อนไขต่างๆในการจัดวางบรรจุภัณฑ์ ในตัวของโปรแกรมเพื่อทำให้เกิดหลักเกณฑ์ใน การจัดวางได้อย่างเป็นระบบ โดยมีหลักการจัดวางบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ ดังต่อไปนี้

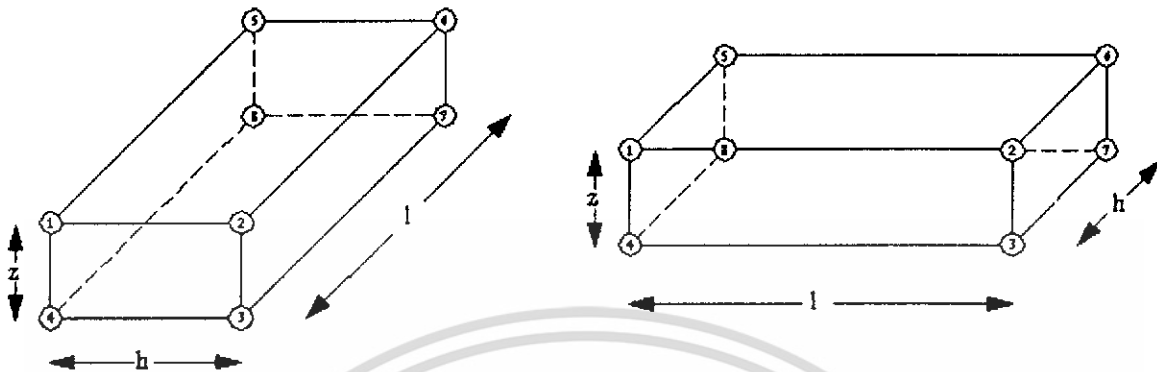
1. สร้างวิธีการ (การจัดเรียง) ที่เป็นไปได้
2. เลือกวิธีการที่ดีที่สุด (ปริมาตรของตู้คอนเทนเนอร์ได้คุ้มค่าที่สุด)
3. วิธีการที่ดีว่าดีที่สุดคือ วิธีที่จัดเรียงบรรจุภัณฑ์ให้มีปริมาตรเหลือน้อยที่สุดซึ่งในที่นี้จะให้ผู้ใช้กำหนด เป็นร้อยละ เพื่อกำหนดขอบเขตของพื้นที่จัดวาง (Search Space)

3.2.2 กำหนดปัจจัยต่างๆที่นำมาพิจารณาในการแก้ปัญหาทางด้านการบรรจุ

1. บรรจุภัณฑ์ที่สามารถวางได้ด้านเดียว โดยปกติแล้วบรรจุภัณฑ์เหล่านั้นจะมีคำเตือนว่า "ด้านนี้ขึ้น (This side up)" บนบรรจุภัณฑ์ชนิดนี้
2. จำนวนบรรจุภัณฑ์ที่กำหนดให้สามารถวางซ้อนกันได้
3. การพิจารณาการวางบรรจุภัณฑ์ซ้อนกันจะต้องมีพื้นฐานไม่เกินบรรจุภัณฑ์ที่ได้ถูกวางไว้แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. บรรจุภัณฑ์ที่กำหนดให้สามารถหมุนได้ 2 ด้านเท่านั้น



รูปที่ 3.1 รูปแสดงการหมุนของกล่องบรรจุภัณฑ์

5. การพิจารณาถึงความสำคัญของบรรจุภัณฑ์ที่จำเป็นต้องส่งก่อน ให้ทำการบรรจุในด้านหน้าสุดของผู้คอนเทนเนอร์

6. สินค้าที่ส่งให้ลูกค้ารายเดียวกันจะถูกจัดวางไว้ชิดกัน

3.3 โครงสร้างในการสร้างโปรแกรมการจัดวางกล่องบรรจุภัณฑ์

โครงสร้างในการสร้างโปรแกรมการจัดวางกล่องบรรจุภัณฑ์จะประกอบด้วย 3 ส่วนหลักๆ ดังต่อไปนี้

3.3.1 ส่วนรับข้อมูลของโปรแกรม จะประกอบด้วย

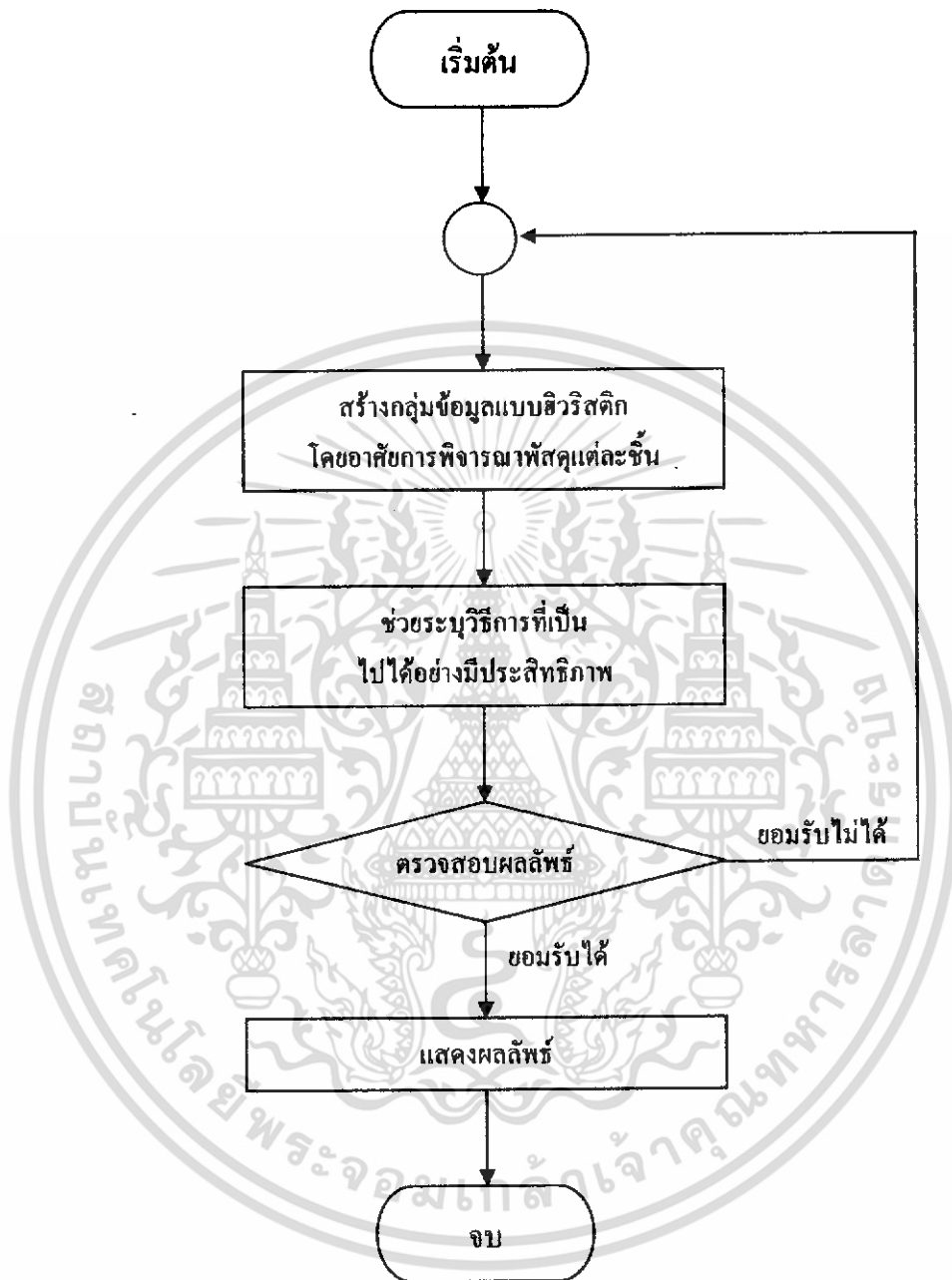
1. ข้อมูลของพื้นที่ระวางบรรจุภัณฑ์ หรือตู้คอนเทนเนอร์

- 1.1 ขนาด (Size) คือ ความกว้าง (W) ความยาว (L) และความสูง (H) ของตู้คอนเทนเนอร์
- 1.2 ความจุ (Volume) คือ ปริมาตรภายในของตู้คอนเทนเนอร์ที่สามารถบรรจุได้
- 1.3 น้ำหนัก (Payload) คือ น้ำหนักของตู้คอนเทนเนอร์ที่สามารถบรรจุได้

2. ข้อมูลของพัสดุ หรือกล่องบรรจุภัณฑ์

- 2.1 ชนิด (Box Type) คือ ชื่อชนิดของสินค้าและกล่องบรรจุภัณฑ์
- 2.2 จำนวน คือ จำนวนกล่องที่ต้องการบรรจุ
- 2.3 ขนาด (Size) คือ ความกว้าง (W) ความยาว (L) และความสูง (H) ของกล่องบรรจุภัณฑ์
- 2.4 น้ำหนัก (Payload) คือ น้ำหนักของกล่องบรรจุภัณฑ์เมื่อมีสินค้าอยู่ภายในกล่อง
- 2.5 ลำดับการจัดส่ง (Priority) คือ การเรียงลำดับความสำคัญในการจัดส่งของกล่องบรรจุภัณฑ์
- 2.6 สี (Color) คือ การกำหนดสีให้บรรจุภัณฑ์แต่ละประเภท

3.3.2 ส่วนการทำงานของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์



รูปที่ 3.2 Flow Chart แสดงขั้นตอนการทำงานของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์

3.3.3 ส่วนการแสดงผลลัพธ์ของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์

สำหรับส่วนของการแสดงผลลัพธ์ของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์ จะประกอบไปด้วย 3 ส่วน ดังต่อไปนี้

1. การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z)
2. ประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์ โดยใช้สมการในการวิเคราะห์โปรแกรมเพื่อหาค่าตำแหน่งที่เหมาะสมของบรรจุภัณฑ์โดยสามารถคำนวณหาปริมาตรที่เหลือ ดังสมการที่ 3.1

$$V_C = V_A - \sum V_B \quad (3.1)$$

สมการในการคำนวณหาเปอร์เซ็นต์ในการใช้ปริมาตรในการจัดวางบรรจุภัณฑ์ ดังสมการที่ 3.2

$$\left[\frac{V_A - V_C}{V_A} \right] \times 100\% \quad (3.2)$$

คำนวณหาน้ำหนักที่บรรจุได้ ดังสมการที่ 3.3

$$W_B = (\sum W_C) \quad (3.3)$$

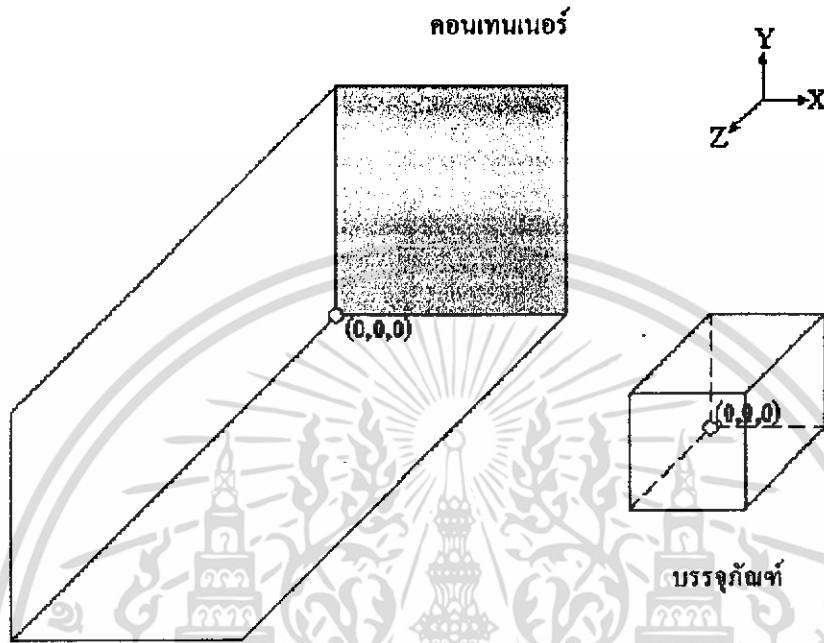
สมการในการคำนวณหาเปอร์เซ็นต์ในการใช้น้ำหนักในการจัดวางบรรจุภัณฑ์ ดังสมการที่ 3.4

$$\left[\frac{W_B}{W_A} \right] \times 100\% \quad (3.4)$$

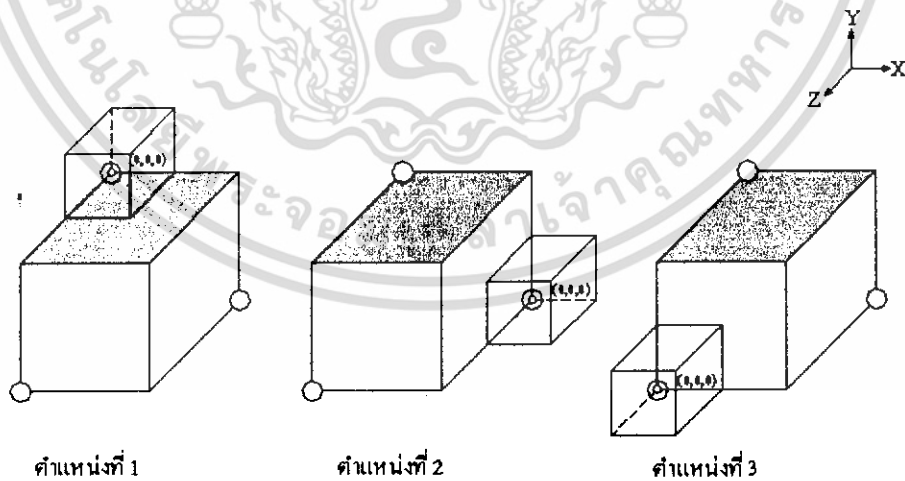
เมื่อ	V_C	= ปริมาตรที่เหลือ
	V_A	= ปริมาตรของพื้นที่จัดวางบรรจุภัณฑ์
	$\sum V_B$	= ผลรวมของปริมาตรบรรจุภัณฑ์ที่ได้ถูกจัดลงไปในพื้นที่จัดวางบรรจุภัณฑ์
	W_B	= น้ำหนักที่บรรจุได้
	W_A	= น้ำหนักที่สามารถรับได้ของตู้คอนเทนเนอร์
	$\sum W_C$	= ผลรวมของน้ำหนักบรรจุภัณฑ์ที่ได้ถูกจัดลงไปในพื้นที่จัดวางบรรจุภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ซึ่งวิธีในการจัดเรียงกล่องบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ ทำโดยการกำหนดจุดพิกัดเริ่มต้นสำหรับตู้คอนเทนเนอร์ และบรรจุภัณฑ์ ดังรูปที่ 3.3, 3.4



รูปที่ 3.3 แสดงตำแหน่งพิกัดที่ (0, 0, 0) ของตู้คอนเทนเนอร์ และบรรจุภัณฑ์



รูปที่ 3.4 แสดงจุดพิกัดที่สามารถทำการวางบรรจุภัณฑ์ขึ้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา ๓๒ ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 องค์ประกอบของโปรแกรมการจัดวางกล่องบรรจุภัณฑ์

โปรแกรมการจัดวางกล่องบรรจุภัณฑ์ได้เขียนขึ้นมาเพื่อช่วยในการทำให้เกิดการจัดวางกล่องบรรจุภัณฑ์ให้มีพื้นที่สูญเปล่าน้อยที่สุดและเพื่อนำมาเป็นเครื่องมือช่วยในการตัดสินใจการจัดวางบรรจุภัณฑ์ในรูปแบบ 3 มิติซึ่งสามารถแบ่งองค์ประกอบในส่วนย่อยๆ ได้ ดังต่อไปนี้

3.4.1 ส่วนการรับข้อมูลของโปรแกรม

ในส่วนนี้จะเกี่ยวข้องกับกรรับข้อมูลที่จำเป็นต่อการจัดวางกล่องบรรจุภัณฑ์ของพื้นที่ระวางบรรจุภัณฑ์หรือตู้คอนเทนเนอร์ และ พัสดุ หรือกล่องบรรจุภัณฑ์ เช่น ชื่อ, ขนาด, ความจุ, น้ำหนัก, จำนวน และ ลำดับการขนส่ง เป็นต้น ดังรูปที่ 3.5

ลำดับ	พัสดุ	ความยาว	ความกว้าง	ความสูง	น้ำหนัก	จำนวน	ลำดับ	จำนวน
1	พัสดุทดสอบ หมายเลข 1	5	29	61	42	5	1	1
2	พัสดุทดสอบ หมายเลข 2	10	38	24	41	30	5	2
3	พัสดุทดสอบ หมายเลข 3	10	36	23	46	25	7	3
4	พัสดุทดสอบ หมายเลข 4	4	35	22	41	10	3	4
5	พัสดุทดสอบ หมายเลข 5	8	29	22	43	15	8	5
6	พัสดุทดสอบ หมายเลข 6	5	28	21	44	15	6	6
7	พัสดุทดสอบ หมายเลข 7	5	29	21	53	20	9	7
8	พัสดุทดสอบ หมายเลข 8	2	37	20	47	10	10	8
9	พัสดุทดสอบ หมายเลข 9	6	36	20	51	30	2	9
10	พัสดุทดสอบ หมายเลข 10	5	30	17	64	25	4	10

รูปที่ 3.5 แสดงส่วนการรับข้อมูลของโปรแกรม

3.4.2 ส่วนการแสดงผลลัพธ์ของโปรแกรม

ในส่วนของการแสดงผลลัพธ์ของโปรแกรม จะประกอบไปด้วย 3 ส่วนดังต่อไปนี้

1. การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ดังรูปที่ 3.6

โปรแกรมบรรจุหีบห่อภัณฑ์ รุ่น 1.0

File	Solved	Help
1	หีบห่อกล่อง หมายเลข 1	0 0 0 61 29 42 1
2	หีบห่อกล่อง หมายเลข 1	61 0 0 61 29 42 1
3	หีบห่อกล่อง หมายเลข 1	122 0 0 61 29 42 1
4	หีบห่อกล่อง หมายเลข 1	122 29 0 61 29 42 1
5	หีบห่อกล่อง หมายเลข 1	122 58 0 61 29 42 1
6	หีบห่อกล่อง หมายเลข 9	122 87 0 51 36 20 2
7	หีบห่อกล่อง หมายเลข 9	122 123 0 51 36 20 2
8	หีบห่อกล่อง หมายเลข 9	122 159 0 51 36 20 2
9	หีบห่อกล่อง หมายเลข 9	61 29 0 51 36 20 2
10	หีบห่อกล่อง หมายเลข 9	61 65 0 51 36 20 2
11	หีบห่อกล่อง หมายเลข 9	61 101 0 51 36 20 2
12	หีบห่อกล่อง หมายเลข 4	61 137 0 41 35 22 3
13	หีบห่อกล่อง หมายเลข 4	0 29 0 41 35 22 3
14	หีบห่อกล่อง หมายเลข 4	0 64 0 41 35 22 3
15	หีบห่อกล่อง หมายเลข 4	0 99 0 41 35 22 3
16	หีบห่อกล่อง หมายเลข 2	0 134 0 41 38 24 5
17	หีบห่อกล่อง หมายเลข 10	0 0 42 64 30 17 4
18	หีบห่อกล่อง หมายเลข 10	64 0 42 64 30 17 4
19	หีบห่อกล่อง หมายเลข 10	128 0 42 64 30 17 4

22/5/2549 21:57

รูปที่ 3.6 แสดงตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z)

2. ประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์ ดังรูปที่ 3.7

โปรแกรมบรรจุหีบห่อภัณฑ์ รุ่น 1.0

ปริมาณของกระดาษ	32,000,000.00	จบ ชม.
ปริมาณที่ใช้ในการบรรจุ	4,110,165	จบ ชม.
ปริมาณคงเหลือ	27,889,840	จบ ชม.
อัตราการใช้ปริมาณในการบรรจุโดยเฉลี่ย	12.84	
จำนวนก้อนพวงในการคำนวณ	6	

22/5/2549 21:58

รูปที่ 3.7 แสดงประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ดังรูปที่ 3.8



รูปที่ 3.8 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงานและการทดลอง

ในบทนี้จะเป็นการแสดงผลของการทดลองจากโปรแกรมการจัดวางกล่องบรรจุภัณฑ์ที่ผู้ดำเนินงานจัดทำขึ้น นำมาทดลองใช้โปรแกรมเพื่อหาวิธีการจัดวางกล่องบรรจุภัณฑ์ ได้ดังต่อไปนี้

4.1 ทดสอบโปรแกรมการจัดวางกล่องบรรจุภัณฑ์

กรณีศึกษาของปัญหาการบรรจุตู้คอนเทนเนอร์ที่ได้ทำการศึกษาประกอบด้วยข้อมูล 4 ชุด ดังตารางที่ 4.1, 4.2, 4.3 และ 4.4 โดยจัดวางในขนาดพื้นที่จัดเก็บขนาด (กว้าง = 100, ยาว = 100 และลึก = 300) แต่แตกต่างกันตรงลำดับการจัดส่งเพื่อหาข้อเปรียบเทียบในการจัดวางของโปรแกรม โดยให้มีการจัดเรียงกล่องเพื่อดูประสิทธิภาพของการจัดเรียงว่าในการจัดเรียง แบบใดใน 4 แบบ ของเงื่อนไขที่ได้กำหนดขึ้นว่าจะให้ประสิทธิภาพของการจัดเรียงสูงสุด มี 4 วิธีการ ตามจำนวนชุดของข้อมูล ดังต่อไปนี้

1. เรียงจากกล่องใหญ่ตามด้วยกล่องกลาง และสุดท้ายคือกล่องเล็ก ดังแสดงข้อมูลในตารางที่ 4.1
2. เรียงจากกล่องกลางตามด้วยกล่องเล็ก และสุดท้ายคือกล่องใหญ่ ดังแสดงข้อมูลในตารางที่ 4.2
3. เรียงจากกล่องเล็กตามด้วยกล่องใหญ่ และสุดท้ายคือกล่องกลาง ดังแสดงข้อมูลในตารางที่ 4.3
4. การจัดเรียงกล่องโดยแบบสุ่ม โดยไม่กำหนดว่าให้กล่องขนาดใดเข้าก่อนและหลัง ดังแสดงข้อมูลในตารางที่ 4.4

ตำหนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ตารางที่ 4.1 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 1

ลำดับ	ชื่อ	จำนวน	ความกว้าง	ความสูง	ความลึก	น้ำหนัก	ลำดับการจัดส่ง	สี
1	กล่องขนาดใหญ่ 1	5	29.5	42	61.3	40	1	1
2	กล่องขนาดใหญ่ 2	8	38	41.7	24.2	35	1	2
3	กล่องขนาดใหญ่ 3	7	36	46.5	23.8	38	1	3
4	กล่องขนาดใหญ่ 4	5	35	41.5	22.7	42	1	4
5	กล่องขนาดกลาง 1	6	29	43.7	22	30	2	5
6	กล่องขนาดกลาง 2	4	28.5	44	21.5	25	2	6
7	กล่องขนาดกลาง 3	6	29.5	53.5	21.5	27	2	7
8	กล่องขนาดกลาง 4	7	37	47.5	20.2	32	2	8
9	กล่องขนาดเล็ก 1	5	31.5	53	20.1	23	3	9
10	กล่องขนาดเล็ก 2	2	36	51	17.7	15	3	10
11	กล่องขนาดเล็ก 3	2	30	43.5	17.5	22	3	11
12	กล่องขนาดเล็ก 4	3	52.3	64.5	8	18	3	12

ตารางที่ 4.2 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 2

ลำดับ	ชื่อ	จำนวน	ความกว้าง	ความสูง	ความลึก	น้ำหนัก	ลำดับการจัดส่ง	สี
1	กล่องขนาดใหญ่ 1	5	29.5	42	61.3	40	3	1
2	กล่องขนาดใหญ่ 2	8	38	41.7	24.2	35	3	2
3	กล่องขนาดใหญ่ 3	7	36	46.5	23.8	38	3	3
4	กล่องขนาดใหญ่ 4	5	35	41.5	22.7	42	3	4
5	กล่องขนาดกลาง 1	6	29	43.7	22	30	1	5
6	กล่องขนาดกลาง 2	4	28.5	44	21.5	25	1	6
7	กล่องขนาดกลาง 3	6	29.5	53.5	21.5	27	1	7
8	กล่องขนาดกลาง 4	7	37	47.5	20.2	32	1	8
9	กล่องขนาดเล็ก 1	5	31.5	53	20.1	23	2	9
10	กล่องขนาดเล็ก 2	2	36	51	17.7	15	2	10
11	กล่องขนาดเล็ก 3	2	30	43.5	17.5	22	2	11
12	กล่องขนาดเล็ก 4	3	52.3	64.5	8	18	2	12

62759

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 3

ลำดับ	ชื่อ	จำนวน	ความกว้าง	ความสูง	ความลึก	น้ำหนัก	ลำดับการจัดส่ง	สี
1	กล่องขนาดใหญ่ 1	5	29.5	42	61.3	40	2	1
2	กล่องขนาดใหญ่ 2	8	38	41.7	24.2	35	2	2
3	กล่องขนาดใหญ่ 3	7	36	46.5	23.8	38	2	3
4	กล่องขนาดใหญ่ 4	5	35	41.5	22.7	42	2	4
5	กล่องขนาดกลาง 1	6	29	43.7	22	30	3	5
6	กล่องขนาดกลาง 2	4	28.5	44	21.5	25	3	6
7	กล่องขนาดกลาง 3	6	29.5	53.5	21.5	27	3	7
8	กล่องขนาดกลาง 4	7	37	47.5	20.2	32	3	8
9	กล่องขนาดเล็ก 1	5	31.5	53	20.1	23	1	9
10	กล่องขนาดเล็ก 2	2	36	51	17.7	15	1	10
11	กล่องขนาดเล็ก 3	2	30	43.5	17.5	22	1	11
12	กล่องขนาดเล็ก 4	3	52.3	64.5	8	18	1	12

ตารางที่ 4.4 ข้อมูลของกล่องบรรจุภัณฑ์ชุดที่ 4

ลำดับ	ชื่อ	จำนวน	ความกว้าง	ความสูง	ความลึก	น้ำหนัก	ลำดับการจัดส่ง	สี
1	กล่องขนาดใหญ่ 1	5	29.5	42	61.3	40	1	1
2	กล่องขนาดใหญ่ 2	8	38	41.7	24.2	35	1	2
3	กล่องขนาดใหญ่ 3	7	36	46.5	23.8	38	1	3
4	กล่องขนาดใหญ่ 4	5	35	41.5	22.7	42	1	4
5	กล่องขนาดกลาง 1	6	29	43.7	22	30	1	5
6	กล่องขนาดกลาง 2	4	28.5	44	21.5	25	1	6
7	กล่องขนาดกลาง 3	6	29.5	53.5	21.5	27	1	7
8	กล่องขนาดกลาง 4	7	37	47.5	20.2	32	1	8
9	กล่องขนาดเล็ก 1	5	31.5	53	20.1	23	1	9
10	กล่องขนาดเล็ก 2	2	36	51	17.7	15	1	10
11	กล่องขนาดเล็ก 3	2	30	43.5	17.5	22	1	11
12	กล่องขนาดเล็ก 4	3	52.3	64.5	8	18	1	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z)

จากการนำข้อมูลของกรณีศึกษาลงในส่วนการรับข้อมูลของโปรแกรม จะมีส่วนแสดงผลลัพธ์ของโปรแกรม ในการบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ดังแสดงในตารางที่ 4.5, 4.6, 4.7 และ 4.8

ตารางที่ 4.5 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 1

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
1	กล่องขนาดใหญ่ 1	0	0	0	29.5	42	61.3
2	กล่องขนาดใหญ่ 2	61.3	0	0	38	41.7	24.2
3	กล่องขนาดใหญ่ 2	61.3	38	0	38	41.7	24.2
4	กล่องขนาดใหญ่ 1	0	29.5	0	29.5	42	61.3
5	กล่องขนาดใหญ่ 1	0	59	0	29.5	42	61.3
6	กล่องขนาดใหญ่ 1	0	0	42	29.5	42	61.3
7	กล่องขนาดใหญ่ 2	61.3	0	42	38	41.7	24.2
8	กล่องขนาดใหญ่ 2	61.3	38	42	38	41.7	24.2
9	กล่องขนาดใหญ่ 1	0	29.5	42	29.5	42	61.3
10	กล่องขนาดใหญ่ 3	0	59	42	36	46.5	23.8
11	กล่องขนาดใหญ่ 3	0	0	84	36	46.5	23.8
12	กล่องขนาดใหญ่ 3	46.5	0	84	36	46.5	23.8
13	กล่องขนาดใหญ่ 3	46.5	36	84	36	46.5	23.8
14	กล่องขนาดใหญ่ 3	0	36	84	36	46.5	23.8
15	กล่องขนาดใหญ่ 3	0	0	107.8	36	46.5	23.8
16	กล่องขนาดใหญ่ 3	46.5	0	107.8	36	46.5	23.8
17	กล่องขนาดใหญ่ 4	46.5	36	107.8	35	41.5	22.7
18	กล่องขนาดใหญ่ 4	0	36	107.8	35	41.5	22.7
19	กล่องขนาดใหญ่ 2	0	0	131.6	38	41.7	24.2
20	กล่องขนาดใหญ่ 2	41.7	0	131.6	38	41.7	24.2
21	กล่องขนาดใหญ่ 2	41.7	38	131.6	38	41.7	24.2
22	กล่องขนาดใหญ่ 2	0	38	131.6	38	41.7	24.2
23	กล่องขนาดใหญ่ 4	0	0	155.8	35	41.5	22.7
24	กล่องขนาดใหญ่ 4	41.5	0	155.8	35	41.5	22.7
25	กล่องขนาดใหญ่ 4	41.5	35	155.8	35	41.5	22.7
26	กล่องขนาดกลาง 3	0	0	178.5	29.5	53.5	21.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 1 (ต่อ)

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
27	กล่องขนาดกลาง 2	53.5	0	178.5	28.5	44	21.5
28	กล่องขนาดกลาง 2	53.5	28.5	178.5	28.5	44	21.5
29	กล่องขนาดกลาง 2	53.5	57	178.5	28.5	44	21.5
30	กล่องขนาดกลาง 3	0	29.5	178.5	29.5	53.5	21.5
31	กล่องขนาดกลาง 3	0	59	178.5	29.5	53.5	21.5
32	กล่องขนาดกลาง 3	0	0	200	29.5	53.5	21.5
33	กล่องขนาดกลาง 2	53.5	0	200	28.5	44	21.5
34	กล่องขนาดเล็ก 3	53.5	28.5	200	30	43.5	17.5
35	กล่องขนาดกลาง 3	0	29.5	200	29.5	53.5	21.5
36	กล่องขนาดกลาง 3	0	59	200	29.5	53.5	21.5
37	กล่องขนาดกลาง 4	0	0	221.5	37	47.5	20.2
38	กล่องขนาดกลาง 4	47.5	0	221.5	37	47.5	20.2
39	กล่องขนาดกลาง 4	47.5	37	221.5	37	47.5	20.2
40	กล่องขนาดกลาง 4	0	37	221.5	37	47.5	20.2
41	กล่องขนาดกลาง 4	0	0	241.7	37	47.5	20.2
42	กล่องขนาดกลาง 4	47.5	0	241.7	37	47.5	20.2
43	กล่องขนาดกลาง 4	47.5	37	241.7	37	47.5	20.2
44	กล่องขนาดเล็ก 3	0	37	241.7	30	43.5	17.5
45	กล่องขนาดกลาง 1	0	0	261.9	29	43.7	22
46	กล่องขนาดกลาง 1	43.7	0	261.9	29	43.7	22
47	กล่องขนาดกลาง 1	43.7	29	261.9	29	43.7	22
48	กล่องขนาดกลาง 1	43.7	58	261.9	29	43.7	22
49	กล่องขนาดกลาง 1	0	29	261.9	29	43.7	22
50	กล่องขนาดกลาง 1	0	58	261.9	29	43.7	22
51	กล่องขนาดเล็ก 4	0	0	283.9	52.3	64.5	8
52	กล่องขนาดเล็ก 4	0	0	291.9	52.3	64.5	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 2

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
1	กล่องขนาดกลาง 3	0	0	0	29.5	53.5	21.5
2	กล่องขนาดกลาง 2	53.5	0	0	28.5	44	21.5
3	กล่องขนาดกลาง 2	53.5	28.5	0	28.5	44	21.5
4	กล่องขนาดกลาง 2	53.5	57	0	28.5	44	21.5
5	กล่องขนาดกลาง 3	0	29.5	0	29.5	53.5	21.5
6	กล่องขนาดกลาง 3	0	59	0	29.5	53.5	21.5
7	กล่องขนาดกลาง 3	0	0	21.5	29.5	53.5	21.5
8	กล่องขนาดกลาง 2	53.5	0	21.5	28.5	44	21.5
9	กล่องขนาดเล็ก 3	53.5	28.5	21.5	30	43.5	17.5
10	กล่องขนาดกลาง 3	0	29.5	21.5	29.5	53.5	21.5
11	กล่องขนาดกลาง 3	0	59	21.5	29.5	53.5	21.5
12	กล่องขนาดกลาง 4	0	0	43	37	47.5	20.2
13	กล่องขนาดกลาง 4	47.5	0	43	37	47.5	20.2
14	กล่องขนาดกลาง 4	47.5	37	43	37	47.5	20.2
15	กล่องขนาดกลาง 4	0	37	43	37	47.5	20.2
16	กล่องขนาดกลาง 4	0	0	63.2	37	47.5	20.2
17	กล่องขนาดกลาง 4	47.5	0	63.2	37	47.5	20.2
18	กล่องขนาดกลาง 4	47.5	37	63.2	37	47.5	20.2
19	กล่องขนาดเล็ก 3	0	37	63.2	30	43.5	17.5
20	กล่องขนาดกลาง 1	0	0	83.4	29	43.7	22
21	กล่องขนาดกลาง 1	43.7	0	83.4	29	43.7	22
22	กล่องขนาดกลาง 1	43.7	29	83.4	29	43.7	22
23	กล่องขนาดกลาง 1	43.7	58	83.4	29	43.7	22
24	กล่องขนาดกลาง 1	0	29	83.4	29	43.7	22
25	กล่องขนาดกลาง 1	0	58	83.4	29	43.7	22
26	กล่องขนาดเล็ก 4	0	0	105.4	52.3	64.5	8
27	กล่องขนาดเล็ก 4	0	0	113.5	52.3	64.5	8
28	กล่องขนาดเล็ก 4	0	0	121.4	52.3	64.5	8
29	กล่องขนาดเล็ก 1	0	0	129.4	31.5	53	20.1
30	กล่องขนาดเล็ก 1	0	31.5	129.4	31.5	53	20.1
31	กล่องขนาดเล็ก 1	0	63	129.4	31.5	53	20.1
32	กล่องขนาดเล็ก 1	0	0	149.5	31.5	53	20.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 2 (ต่อ)

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
33	กล่องขนาดเล็ก 1	0	31.5	149.5	31.5	53	20.1
34	กล่องขนาดเล็ก 2	0	63	149.5	36	51	17.7
35	กล่องขนาดเล็ก 2	0	0	169.9	36	51	17.7
36	กล่องขนาดใหญ่ 1	0	0	187.3	29.5	42	61.3
37	กล่องขนาดใหญ่ 2	61.3	0	187.3	38	41.7	24.2
38	กล่องขนาดใหญ่ 2	61.3	38	187.3	38	41.7	24.2
39	กล่องขนาดใหญ่ 1	0	29.5	187.3	29.5	42	61.3
40	กล่องขนาดใหญ่ 1	0	59	187.3	29.5	42	61.3
41	กล่องขนาดใหญ่ 1	0	0	229.3	29.5	42	61.3
42	กล่องขนาดใหญ่ 2	61.3	0	229.3	38	41.7	24.2
43	กล่องขนาดใหญ่ 2	61.3	38	229.3	38	41.7	24.2
44	กล่องขนาดใหญ่ 1	0	29.5	229.3	29.5	42	61.3
45	กล่องขนาดใหญ่ 3	0	59	229.3	36	46.5	23.8
46	กล่องขนาดใหญ่ 3	0	0	271.3	36	46.5	23.8
47	กล่องขนาดใหญ่ 3	46.5	0	271.3	36	46.5	23.8
48	กล่องขนาดใหญ่ 3	46.5	36	271.3	36	46.5	23.8
49	กล่องขนาดใหญ่ 3	0	36	271.3	36	46.5	23.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 3

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
1	กล่องขนาดเล็ก 4	0	0	0	52.3	64.5	8
2	กล่องขนาดเล็ก 4	0	0	8	52.3	64.5	8
3	กล่องขนาดเล็ก 4	0	0	16	52.3	64.5	8
4	กล่องขนาดเล็ก 1	0	0	24	31.5	53	20.1
5	กล่องขนาดเล็ก 3	53	0	24	30	43.5	17.5
6	กล่องขนาดเล็ก 3	53	30	24	30	43.5	17.5
7	กล่องขนาดเล็ก 1	0	31.5	24	31.5	53	20.1
8	กล่องขนาดเล็ก 1	0	63	24	31.5	53	20.1
9	กล่องขนาดเล็ก 1	0	0	44.1	31.5	53	20.1
10	กล่องขนาดเล็ก 1	0	31.5	44.1	31.5	53	20.1
11	กล่องขนาดเล็ก 2	0	63	44.1	36	51	17.7
12	กล่องขนาดเล็ก 2	0	0	64.2	36	51	17.7
13	กล่องขนาดใหญ่ 1	0	0	81.899	29.5	42	61.3
14	กล่องขนาดใหญ่ 2	61.3	0	81.899	38	41.7	24.2
15	กล่องขนาดใหญ่ 2	61.3	38	81.899	38	41.7	24.2
16	กล่องขนาดใหญ่ 1	0	29.5	81.899	29.5	42	61.3
17	กล่องขนาดใหญ่ 1	0	59	81.899	29.5	42	61.3
18	กล่องขนาดใหญ่ 1	0	0	123.9	29.5	42	61.3
19	กล่องขนาดใหญ่ 2	61.3	0	123.9	38	41.7	24.2
20	กล่องขนาดใหญ่ 2	61.3	38	123.9	38	41.7	24.2
21	กล่องขนาดใหญ่ 1	0	29.5	123.9	29.5	42	61.3
22	กล่องขนาดใหญ่ 3	0	59	123.9	36	46.5	23.8
23	กล่องขนาดใหญ่ 3	0	0	165.9	36	46.5	23.8
24	กล่องขนาดใหญ่ 3	46.5	0	165.9	36	46.5	23.8
25	กล่องขนาดใหญ่ 3	46.5	36	165.9	36	46.5	23.8
26	กล่องขนาดใหญ่ 3	0	36	165.9	36	46.5	23.8
27	กล่องขนาดใหญ่ 3	0	0	189.7	36	46.5	23.8
28	กล่องขนาดใหญ่ 3	46.5	0	189.7	36	46.5	23.8
29	กล่องขนาดใหญ่ 4	46.5	36	189.7	35	41.5	22.7
30	กล่องขนาดใหญ่ 4	0	36	189.7	35	41.5	22.7
31	กล่องขนาดใหญ่ 2	0	0	213.5	38	41.7	24.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา หรือต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 3 (ต่อ)

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
32	กล่องขนาดใหญ่ 2	41.7	0	213.5	38	41.7	24.2
33	กล่องขนาดใหญ่ 2	41.7	38	213.5	38	41.7	24.2
34	กล่องขนาดใหญ่ 2	0	38	213.5	38	41.7	24.2
35	กล่องขนาดใหญ่ 4	0	0	237.7	35	41.5	22.7
36	กล่องขนาดใหญ่ 4	41.5	0	237.7	35	41.5	22.7
37	กล่องขนาดใหญ่ 4	41.5	35	237.7	35	41.5	22.7
38	กล่องขนาดใหญ่ 4	0	0	260.4	35	41.5	22.7
39	กล่องขนาดกลาง 2	53.5	0	260.4	28.5	44	21.5
40	กล่องขนาดกลาง 2	53.5	28.5	260.4	28.5	44	21.5
41	กล่องขนาดกลาง 2	53.5	57	260.4	28.5	44	21.5
42	กล่องขนาดกลาง 3	0	29.5	260.4	29.5	53.5	21.5
43	กล่องขนาดกลาง 3	0	59	260.4	29.5	53.5	21.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ 4

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
1	กล่องขนาดเล็ก 4	0	0	0	52.3	64.5	8
2	กล่องขนาดเล็ก 4	0	0	8	52.3	64.5	8
3	กล่องขนาดเล็ก 4	0	0	16	52.3	64.5	8
4	กล่องขนาดใหญ่ 1	0	0	24	29.5	42	61.3
5	กล่องขนาดใหญ่ 2	61.3	0	24	38	41.7	24.2
6	กล่องขนาดใหญ่ 2	61.3	38	24	38	41.7	24.2
7	กล่องขนาดใหญ่ 1	0	29.5	24	29.5	42	61.3
8	กล่องขนาดใหญ่ 1	0	59	24	29.5	42	61.3
9	กล่องขนาดใหญ่ 1	0	0	66	29.5	42	61.3
10	กล่องขนาดใหญ่ 2	61.3	0	66	38	41.7	24.2
11	กล่องขนาดใหญ่ 2	61.3	38	66	38	41.7	24.2
12	กล่องขนาดใหญ่ 1	0	29.5	66	29.5	42	61.3
13	กล่องขนาดกลาง 3	0	59	66	29.5	53.5	21.5
14	กล่องขนาดกลาง 3	0	0	108	29.5	53.5	21.5
15	กล่องขนาดกลาง 2	53.5	0	108	28.5	44	21.5
16	กล่องขนาดกลาง 2	53.5	28.5	108	28.5	44	21.5
17	กล่องขนาดกลาง 2	53.5	57	108	28.5	44	21.5
18	กล่องขนาดกลาง 3	0	29.5	108	29.5	53.5	21.5
19	กล่องขนาดกลาง 3	0	59	108	29.5	53.5	21.5
20	กล่องขนาดกลาง 3	0	0	129.5	29.5	53.5	21.5
21	กล่องขนาดกลาง 2	53.5	0	129.5	28.5	44	21.5
22	กล่องขนาดเล็ก 3	53.5	28.5	129.5	30	43.5	17.5
23	กล่องขนาดเล็ก 3	53.5	58.5	129.5	30	43.5	17.5
24	กล่องขนาดกลาง 3	0	29.5	129.5	29.5	53.5	21.5
25	กล่องขนาดเล็ก 1	0	59.5	129.5	31.5	53	20.1
26	กล่องขนาดเล็ก 1	0	0	151	31.5	53	20.1
27	กล่องขนาดเล็ก 1	0	31.5	151	31.5	53	20.1
28	กล่องขนาดเล็ก 1	0	63	151	31.5	53	20.1
29	กล่องขนาดเล็ก 1	0	0	171.1	31.5	53	20.1
30	กล่องขนาดเล็ก 2	0	31.5	171.1	36	51	17.7
31	กล่องขนาดเล็ก 2	0	0	191.2	36	51	17.7
32	กล่องขนาดกลาง 4	0	0	208.9	37	47.5	20.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.8 แสดงลำดับการจัดวางกล่องบรรจุภัณฑ์แบบเป็นโคออดิเนต (X, Y, Z) ของกรณีศึกษาข้อมูลชุดที่ (ต่อ)

ลำดับ	ชื่อ	X	Y	Z	ความกว้าง	ความสูง	ความลึก
33	กล่องขนาดกลาง 4	47.5	0	208.9	37	47.5	20.2
34	กล่องขนาดกลาง 4	47.5	37	208.9	37	47.5	20.2
35	กล่องขนาดกลาง 4	0	37	208.9	37	47.5	20.2
36	กล่องขนาดกลาง 4	0	0	229.1	37	47.5	20.2
37	กล่องขนาดกลาง 4	47.5	0	229.1	37	47.5	20.2
38	กล่องขนาดกลาง 4	47.5	37	229.1	37	47.5	20.2
39	กล่องขนาดใหญ่ 3	0	0	249.3	36	46.5	23.8
40	กล่องขนาดใหญ่ 3	46.5	0	249.3	36	46.5	23.8
41	กล่องขนาดใหญ่ 3	46.5	36	249.3	36	46.5	23.8
42	กล่องขนาดใหญ่ 3	0	36	249.3	36	46.5	23.8
43	กล่องขนาดใหญ่ 3	0	0	273.1	36	46.5	23.8
44	กล่องขนาดใหญ่ 3	46.5	0	273.1	36	46.5	23.8
45	กล่องขนาดใหญ่ 3	46.5	36	273.1	36	46.5	23.8
46	กล่องขนาดกลาง 1	0	36	273.1	29	43.7	22
47	กล่องขนาดกลาง 1	0	65	273.1	29	43.7	22

4.3 ประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์

จากการนำข้อมูลของกรณีศึกษาลงในส่วนการรับข้อมูลของโปรแกรม ซึ่งจะมีส่วนแสดงผลลัพธ์ของโปรแกรม ในการบอกประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์ ดังแสดงในตารางที่ 4.9

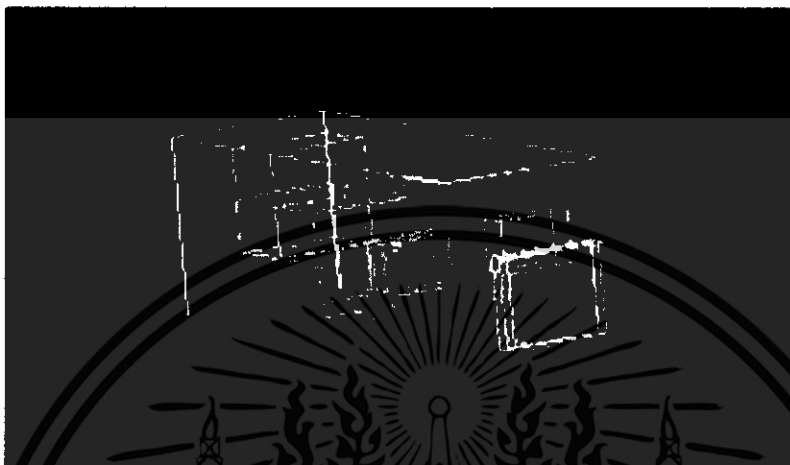
ตารางที่ 4.9 แสดงประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์

ประสิทธิภาพของการจัดวาง	ข้อมูลชุดที่ 1	ข้อมูลชุดที่ 2	ข้อมูลชุดที่ 3	ข้อมูลชุดที่ 4
ปริมาตรของระวาง	3,000,000ลบ.ซม	3,000,000ลบ.ซม	3,000,000ลบ.ซม	3,000,000ลบ.ซม
ปริมาตรที่ใช้ในการบรรจุ	2,110,060ลบ.ซม	1,900,754ลบ.ซม	1,974,512ลบ.ซม	1,913,098ลบ.ซม
ปริมาตรคงเหลือ	889,940ลบ.ซม	1,099,246ลบ.ซม	1,025,488ลบ.ซม	1,086,902ลบ.ซม
อัตราการใช้ปริมาตรในการบรรจุ	70.34	63.36	65.82	63.77
จำนวนกำแพงในการคำนวณ	13	14	13	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ

จากการนำข้อมูลของกรณีศึกษาลงในส่วนการรับข้อมูลของโปรแกรม ซึ่งจะมีส่วนแสดงผลลัพธ์ของโปรแกรม ในการแสดงตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ดังแสดงในรูปที่ 4.1, 4.2, 4.3 และ 4.4

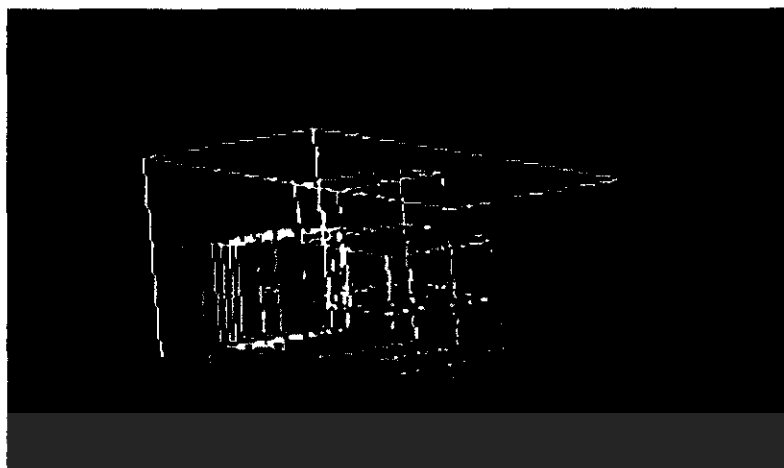


รูปที่ 4.1 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 1



รูปที่ 4.2 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 3



รูปที่ 4.4 แสดงการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ของข้อมูลชุดที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงาน

5.1 การสรุปผลการดำเนินงาน

ปริญญานิพนธ์ฉบับนี้จัดทำขึ้นเพื่อช่วยในการตัดสินใจการจัดวางกล่องบรรจุภัณฑ์ในรูปแบบ 3 มิติ โดยใช้วิธีเชิงรวม (Hybrid Approach) และวิธีทางทฤษฎี (Theoretical Approach) มาการสร้างเงื่อนไขของการจัดวางกล่องบรรจุภัณฑ์ที่เป็นไปได้ในการจัดวางกล่องบรรจุภัณฑ์พื้นฐาน แล้วใช้การทำงานแบบวิธีฮิวริสติก (Heuristic Approach) ในการแก้ปัญหาการบรรจุตู้คอนเทนเนอร์แบบสามมิติ มาใช้สร้างเงื่อนไขของโปรแกรมในการจัดวางกล่องบรรจุภัณฑ์ ซึ่งมีจุดมุ่งหมายที่สำคัญ คือ เพื่อหาวิธีในการจัดวางกล่องบรรจุภัณฑ์ให้มีปริมาตรสูญเปล่าน้อยที่สุด และเพื่อช่วยในการลดต้นทุนการขนส่งหรือขนย้าย โดยใช้ความสามารถในการจัดเก็บฐานข้อมูลและการประมวลผลที่รวดเร็วและแม่นยำของระบบคอมพิวเตอร์เข้ามาช่วยในการคำนวณ และแสดงผลการจัดวางออกมาในรูปแบบกราฟิก 3 มิติ เพื่อให้เข้าใจรูปแบบของการจัดวางได้ง่ายยิ่งขึ้น และเกิดความรวดเร็วในการจัดวางบรรจุภัณฑ์ในกรณีที่มีจำนวนกล่องมากขึ้น โดยเปรียบเทียบจากการจัดแบบคนจัดในแบบเดิม

จากการทดลองโปรแกรมการจัดวางกล่องบรรจุภัณฑ์ได้สุ่มหาข้อมูลตัวอย่างกล่องบรรจุภัณฑ์มา 12 กล่อง โดยกำหนดความแตกต่างกันระหว่างลำดับการจัดส่งเพื่อทดลองว่าลำดับการจัดส่งมีผลต่อการจัดเรียงหรือไม่ จากผลการทดลองทั้ง 4 แบบได้ว่า

- ผลการทดลองที่ 1. ได้ว่าปริมาตรที่ใช้ในการบรรจุเท่ากับ 2,110,060 ลูกบาศก์เซนติเมตร หรือคิดเป็นร้อยละ 70.34 %
- ผลการทดลองที่ 2. ได้ว่าปริมาตรที่ใช้ในการบรรจุเท่ากับ 1,900,754 ลูกบาศก์เซนติเมตร หรือคิดเป็นร้อยละ 63.36 %
- ผลการทดลองที่ 3. ได้ว่าปริมาตรที่ใช้ในการบรรจุเท่ากับ 1,974,512 ลูกบาศก์เซนติเมตร หรือคิดเป็นร้อยละ 65.82 %
- ผลการทดลองที่ 4. ได้ว่าปริมาตรที่ใช้ในการบรรจุเท่ากับ 1,913,098 ลูกบาศก์เซนติเมตร หรือคิดเป็นร้อยละ 63.77 %

จากผลการทดลองสรุปได้ว่า ลำดับการจัดส่งกล่องบรรจุภัณฑ์นั้นทำให้การใช้พื้นที่ในการจัดรูปแบบการจัดวางกล่องบรรจุภัณฑ์นั้นเปลี่ยนแปลงไป โดยที่กล่องบรรจุภัณฑ์ที่มีขนาดใหญ่ถูกจัดเรียงก่อนก็จะทำให้มีประสิทธิภาพสูงในการใช้พื้นที่บรรจุภัณฑ์เพราะจะสามารถนำกล่องบรรจุภัณฑ์ที่มีขนาดเล็กมาวางซ้อนทับกันได้ แต่ถ้ากล่องบรรจุภัณฑ์ที่มีขนาดเล็กถูกจัดเรียงก่อนจะมีโอกาสที่ทำให้กล่องบรรจุภัณฑ์ที่เหลือมาวางซ้อนทับได้นั้นน้อยลงจึงทำให้มีพื้นที่ว่างหลงเหลืออยู่ภายใน ซึ่งไม่สามารถใช้ประโยชน์จากพื้นที่นั้นได้

5.2 ข้อเสนอแนะ

1. ควรดำเนินการพัฒนาและปรับปรุงโปรแกรมการจัดวางกล่องบรรจุภัณฑ์ โดยกำหนดปัจจัยต่างๆที่นำมาพิจารณาในการแก้ปัญหาทางด้านการบรรจุเพิ่มมากขึ้น เพื่อให้สามารถนำไปใช้งานได้จริงในหลายๆ กรณี
2. ควรดำเนินการพัฒนาและปรับปรุงโปรแกรมการจัดวางกล่องบรรจุภัณฑ์ โดยทำการศึกษาวิธีฮิวริสติกวิธี อาจเข้ามาใช้ในการแก้ปัญหา เพื่อให้ได้ผลลัพธ์ที่ดีมากยิ่งขึ้น

3. ควรดำเนินการพัฒนาส่วนแสดงผลและส่วนติดต่อกับผู้ใช้ให้มีประสิทธิภาพมากขึ้น เพื่อให้ผู้ใช้เกิดความ
สะดวกในการทำงานและสามารถเข้าใจได้ง่าย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา 30 ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

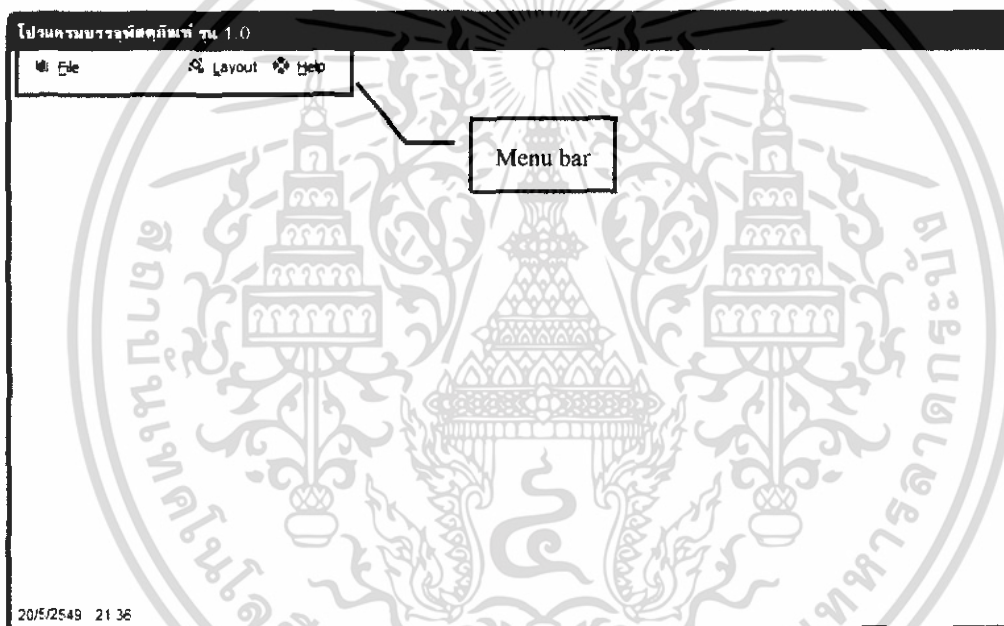
- กิตติ ภัคดีวัฒนกุล, 2546. Visual Basic 6 ฉบับโปรแกรมเมอร์. พิมพ์ครั้งที่ 11. กรุงเทพฯ : บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด.
- คชาเดช วุฒิชัยรังสิต, 2545. วิธีการเลือกประเภทของผลิตภัณฑ์เพื่อบรรจุลงในตู้คอนเทนเนอร์ให้มีมูลค่าผลิตภัณฑ์โดยรวมสูงสุด. วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิชาวิศวกรรมอุตสาหกรรม มหาวิทยาลัยเกษตรศาสตร์.
- ชัยวัฒน์ พิมพ์วัฒน์ , 2543. การพัฒนาขั้นตอนวิธีเชิงพันธุกรรมแบบวิวัฒนาการและการทำงานร่วมกันแบบปรับปรุงสำหรับกรณีศึกษาการบรรจุตู้คอนเทนเนอร์. วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิชาวิศวกรรมการจัดการอุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ.
- โชคชัย โจลานันท์, 2545. ปัญหาการบรรจุในระบบ 3 มิติโดยการพัฒนาวิธีทางฮิวริสติกและการประมวลผลลอจิก. วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิชาวิศวกรรมอุตสาหกรรม มหาวิทยาลัยเกษตรศาสตร์.
- ปริชญ์ สุรศักดิ์ศิลป์, 2545. การหาค่าตอบที่เหมาะสมที่สุดในการจัดเรียงสินค้าเพื่อให้ต้นทุนการเข้าตู้คอนเทนเนอร์ต่ำที่สุด. วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิชาวิศวกรรมอุตสาหกรรม มหาวิทยาลัยเกษตรศาสตร์.
- วัฒน์พงศ์ ภัคดี, 2546. การออกแบบขนาดของบรรจุภัณฑ์และการบรรจุภัณฑ์ลงในตู้คอนเทนเนอร์ให้มีค่าใช้จ่ายต่ำที่สุด. วิทยานิพนธ์ปริญญาโทบริหารธุรกิจ สาขาวิชาวิศวกรรมอุตสาหกรรม มหาวิทยาลัยเกษตรศาสตร์.
- สัจจะ จรัสรุ่งเรือง, 2544. คู่มือการเขียนโปรแกรมและใช้งาน Visual Basic 6.0. พิมพ์ครั้งที่ 5. กรุงเทพฯ : สำนักพิมพ์อินโฟเพรส.

ภาคผนวก ก

คู่มือการใช้งานโปรแกรม

ขั้นตอนการใช้งานของโปรแกรมคอมพิวเตอร์สำหรับการจัดวางกล่องบรรจุภัณฑ์มีขั้นตอนต่างๆ ดังต่อไปนี้

1. ทำการติดตั้งโปรแกรมไว้ในไดเรกทอรีที่สร้างขึ้น
2. ทำการเปิด ไฟล์ที่ชื่อ Bin Packing
3. เมื่อเปิด โปรแกรม จะได้น้ำจอ ดังรูปที่ ผก 1

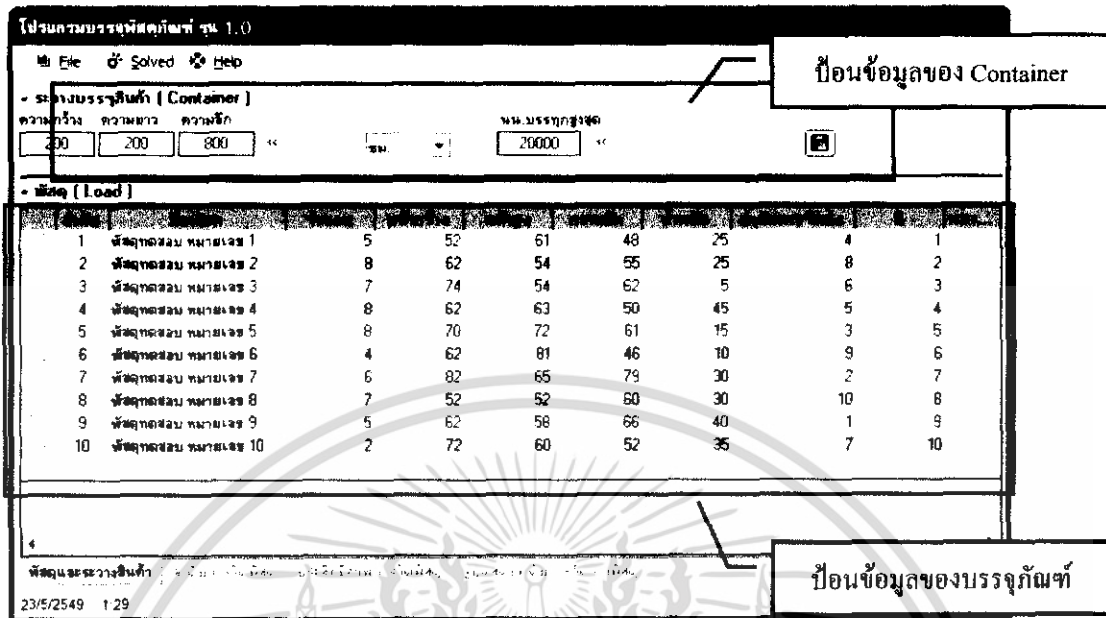


รูปที่ ผก 1 จอหลักของโปรแกรม

Menu bar ใช้เก็บชุดคำสั่งต่างๆ ของโปรแกรมทั้งหมด โดยสามารถคลิกที่ Menu bar ที่จะเรียกใช้งาน ส่วนตัวอักษรทางด้านซ้ายเรียกว่า คีย์ลัด ซึ่งสามารถกดปุ่มบนคีย์บอร์ดแทนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อเกิดไฟล์ใหม่จะพบหน้าจอในส่วนการรับข้อมูลของโปรแกรม ดังรูปที่ ผก 2



รูปที่ ผก 2 แสดงหน้าจอในส่วนการรับข้อมูลของโปรแกรม

ทำการป้อนข้อมูลในส่วนรับข้อมูลของโปรแกรมโดยป้อนข้อมูลที่จำเป็นต่อการจัดวางกล่องบรรจุภัณฑ์ ซึ่งจะประกอบด้วย

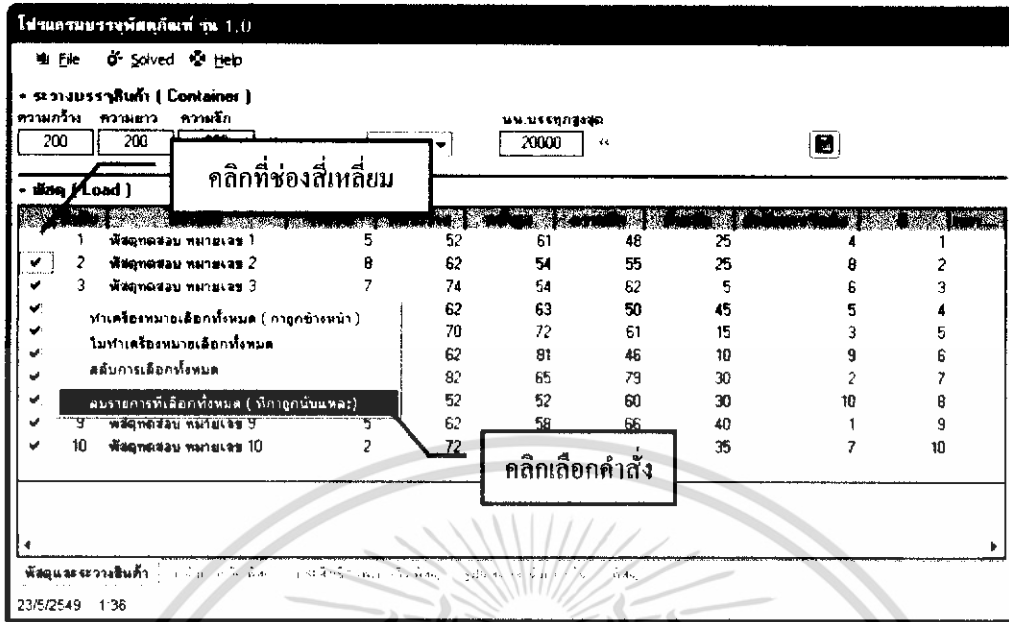
4.1 ข้อมูลของพื้นที่ระวางบรรจุภัณฑ์ หรือตู้คอนเทนเนอร์

- 4.1.1 ขนาด (Size) คือ ความกว้าง (W) ความยาว (L) และความสูง (H) ของตู้คอนเทนเนอร์
- 4.1.2 ความจุ (Volume) คือ ปริมาตรภายในของตู้คอนเทนเนอร์ที่สามารถบรรจุได้
- 4.1.3 น้ำหนัก (Payload) คือ น้ำหนักของตู้คอนเทนเนอร์ที่สามารถบรรจุได้

4.2 ข้อมูลของพัสดุ หรือกล่องบรรจุภัณฑ์

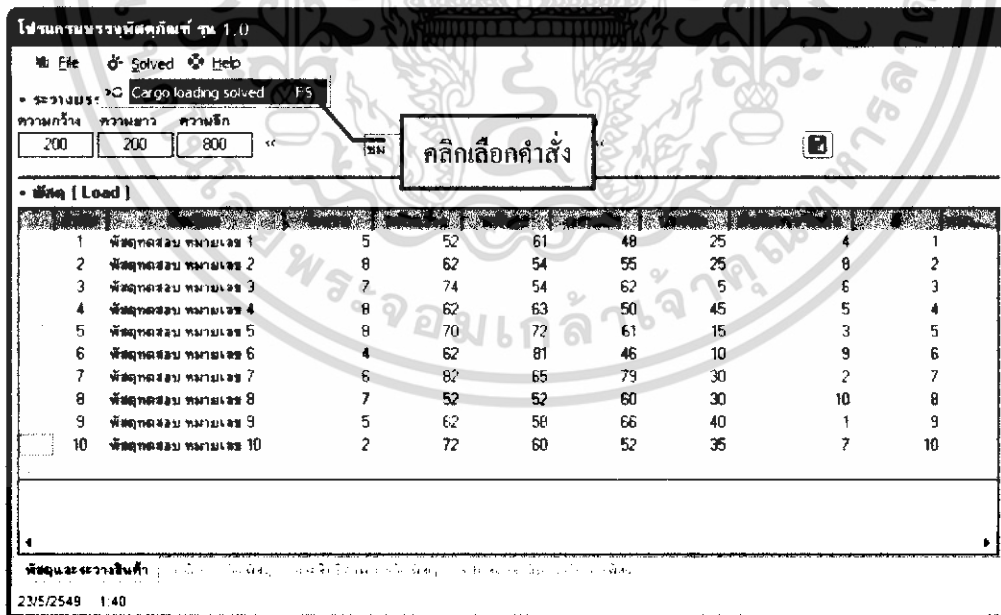
- 4.2.1 ชนิด (Box Type) คือ ชื่อชนิดของสินค้าและกล่องบรรจุภัณฑ์
- 4.2.2 จำนวน คือ จำนวนกล่องที่ต้องการบรรจุ
- 4.2.3 ขนาด (Size) คือ ความกว้าง (W) ความยาว (L) และความสูง (H) ของกล่องบรรจุภัณฑ์
- 4.2.4 น้ำหนัก (Payload) คือ น้ำหนักของกล่องบรรจุภัณฑ์เมื่อมีสินค้าอยู่ภายในกล่อง
- 4.2.5 ลำดับการจัดส่ง (Priority) คือ การเรียงลำดับความสำคัญในการจัดส่งของกล่องบรรจุภัณฑ์

ส่วนเมื่อต้องการเพิ่ม หรือลบจำนวนกล่องบรรจุภัณฑ์ให้คลิกที่ช่องสี่เหลี่ยมหน้าช่องของลำดับกล่องบรรจุภัณฑ์ ต่อมาให้คลิกขวาแล้วเลือกคำสั่งที่ต้องการ ดังรูปที่ ผก 3



รูปที่ ผก 3 แสดงหน้าจอในส่วนการรับข้อมูลของโปรแกรม

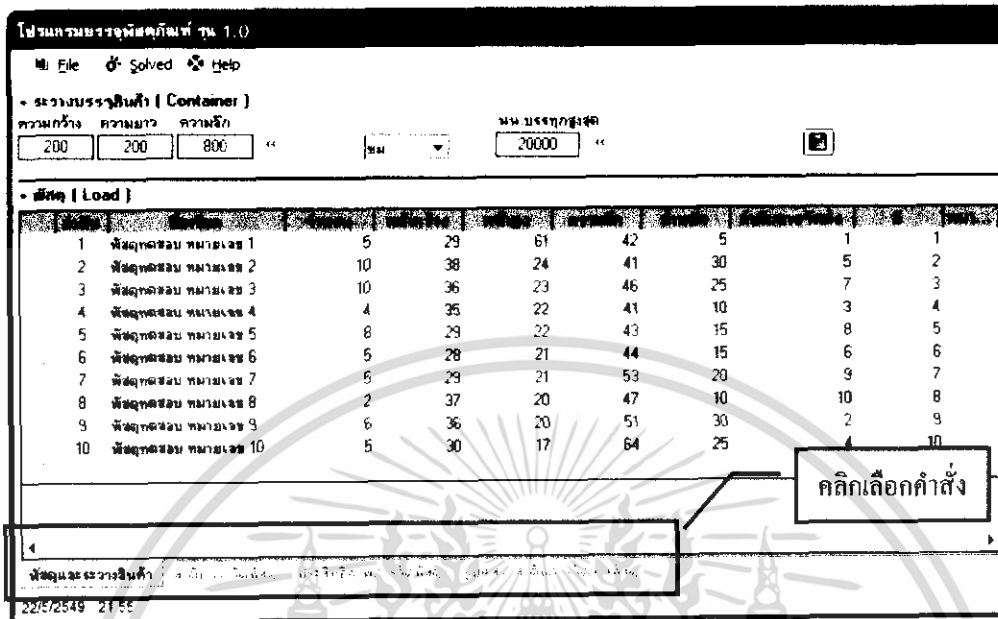
5. เมื่อทำการป้อนข้อมูลในส่วนการรับข้อมูลของโปรแกรมแล้ว ก็ทำการ Solved ข้อมูลเพื่อให้ได้ผลลัพธ์ที่ต้องการ โดยไปที่ Menu bar คลิกที่ Solved เลือกคำสั่งไปที่ Cargo loading Solved ดังรูปที่ ผก 4



รูปที่ ผก 4 แสดงหน้าจอในส่วนการรับข้อมูลของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

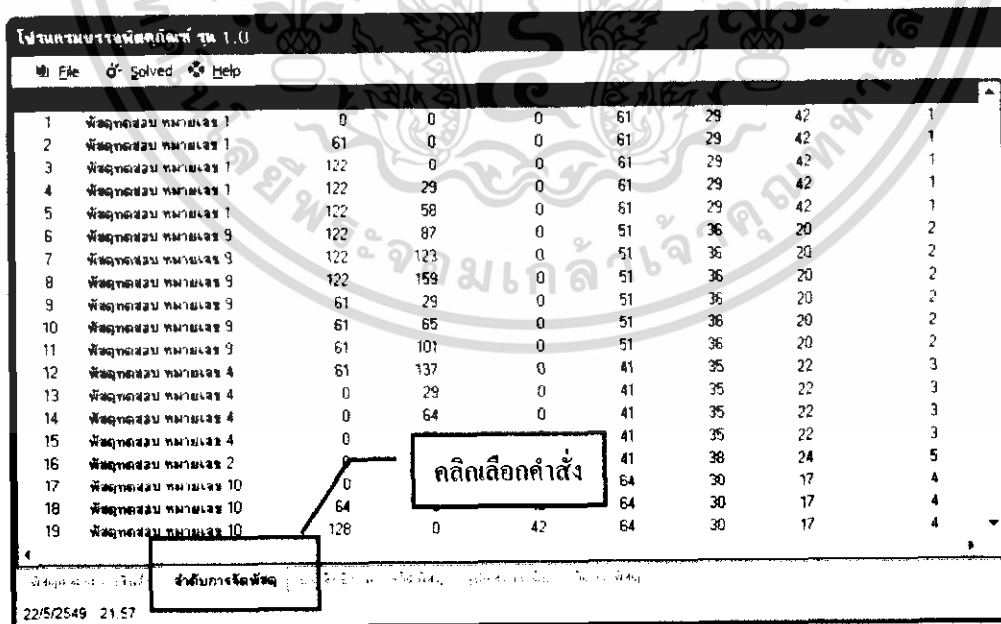
6. เมื่อทำการ Solved ข้อมูลแล้วจะ ได้ส่วนของการแสดงผลลัพธ์ของโปรแกรม โดยทำการเลือกที่ด้านล่างของหน้าจอ ดังรูปที่ ผก 5



รูปที่ ผก 5 แสดงหน้าจอในส่วนการรับข้อมูลของโปรแกรม

ส่วนของการแสดงผลลัพธ์ของโปรแกรมนี้จะประกอบไปด้วย 3 ส่วนดังต่อไปนี้

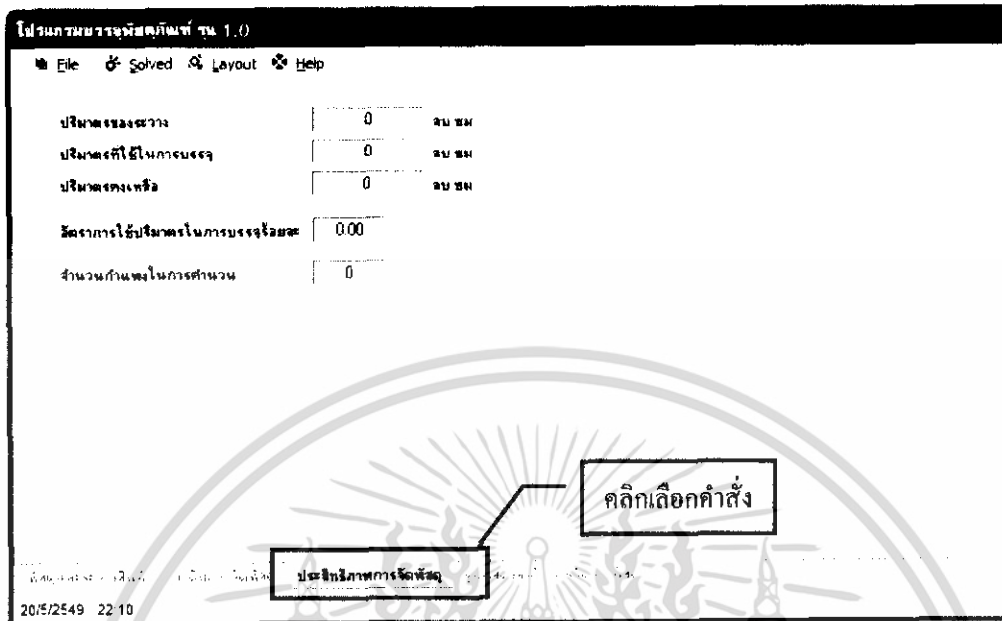
6.1 การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต (X, Y, Z) ดังรูปที่ ผก 6



รูปที่ ผก 6 แสดงหน้าจอบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบเป็น โคออดิเนต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 ประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์ ดังรูปที่ ผก 7



รูปที่ ผก 7 แสดงหน้าจอบอกประสิทธิภาพของการจัดวางกล่องบรรจุภัณฑ์

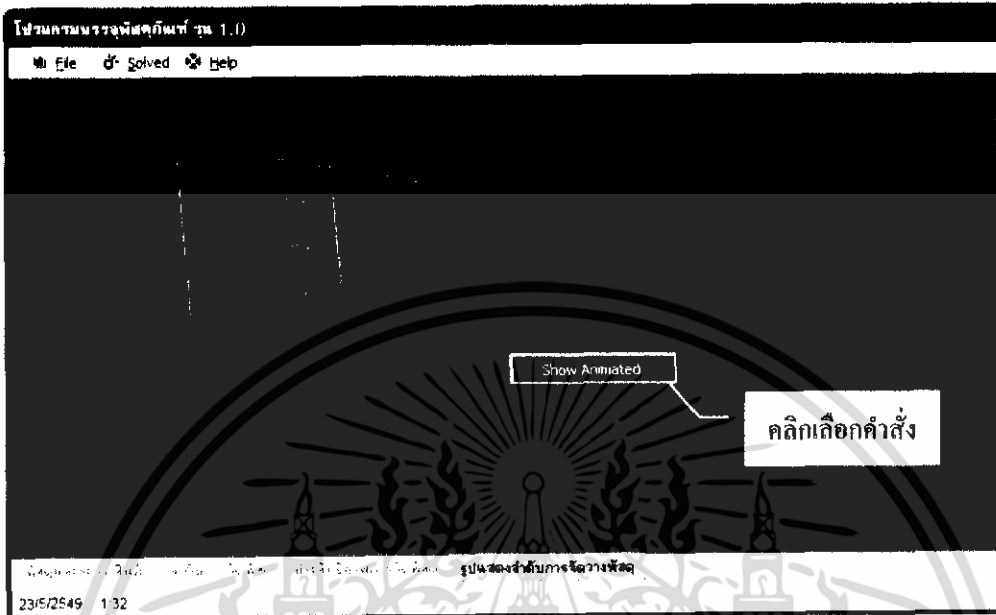
6.3 การบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ดังรูปที่ ผก 8



รูปที่ ผก 8 แสดงหน้าจอบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ

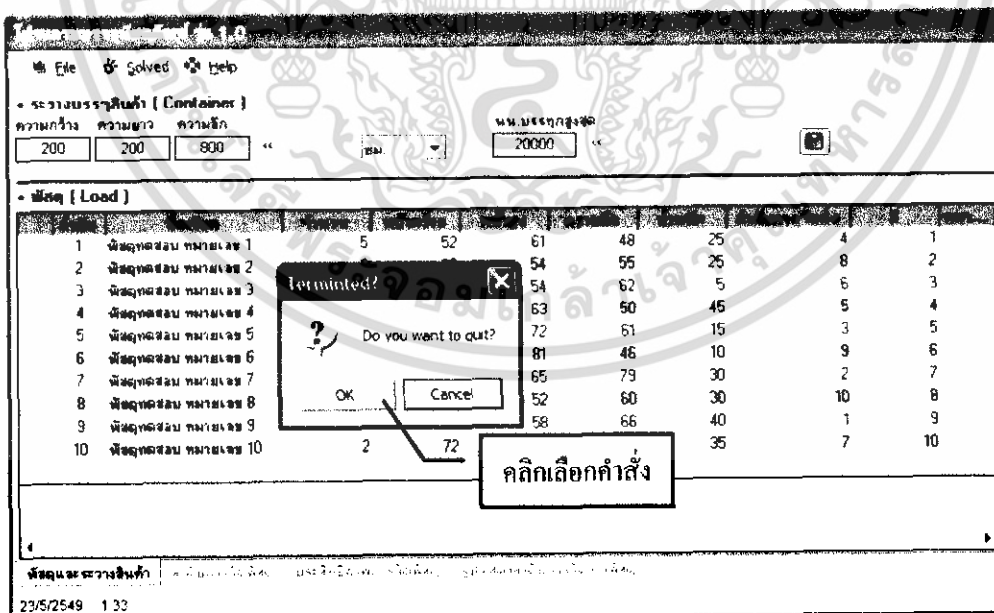
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. เมื่อต้องดูภาพการบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ ที่เป็น Animation ให้คลิกขวาที่หน้าจอบอกตำแหน่งการจัดวางกล่องบรรจุภัณฑ์แบบภาพกราฟิก 3 มิติ แล้วคลิกเลือกคำสั่ง Show Animated ดังรูปที่ ผก 9



รูปที่ ผก 9 แสดงการเลือกคำสั่ง Show Animated

8. เมื่อต้องการออกจากโปรแกรม กดปุ่ม Exit แล้วตอบ OK ดังรูปที่ ผก 10



รูปที่ ผก 10 แสดงการเลือกคำสั่งออกจากโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

โค้ดโปรแกรม

การแสดงผลโค้ดโปรแกรม (Source Code) ในส่วนของรูปแบบการจัดวางของโปรแกรม

Option Explicit

Public CurrentWallID As Long

Private availableVolume As Single

Private iContainerWidth As Single

Private iContainerHeight As Single

Private iContainerDepth As Single

Private iContainerMaxWeight As Single

Private iConsumeSpace As Single

Private iOrder As Long

Private LoadWeight As Single

Private StripeHorizonHeight As Single

Public Type Point3DType

xx As Single

yy As Single

zz As Single

End Type

Private PinPoint As Point3DType

Public Type CoorBoxType

Location As Point3DType

Width As Single

Height As Single

Depth As Single

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

dWidth As Single
dHeight As Single
dDepth As Single
Weight As Single
Priority As Long
Color As OLE_COLOR
End Type

Private iLayer() As CoordBoxType
Private iCurrentLayer As Long

Sub PrepareDb4NewCalculation()

Dim iCon As New ADODB.Connection
Dim iStr As String
Dim Rs As New ADODB.Recordset
Dim tempStr As String
Dim Counter As Long
Dim Captor As Long
Dim xStr As String

On Error Resume Next

iCon.ConnectionString = CreateMsAccessConnectionString(AppsDataSource)

iCon.Open

iForm.grdLoads.Update

DoEvents

iStr = "DROP table t_box_clone"

iCon.Execute iStr

iStr = "DELETE * FROM t_box_load"

iCon.Execute iStr

iStr = "DELETE * FROM t_stripe"

iCon.Execute iStr

iStr = "DELETE * FROM t_box_clone"

iCon.Execute iStr

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

iStr = "SELECT * INTO t_box_clone FROM t_box"
iCon.Execute iStr
iStr = "update t_box_clone set box_enabled=false"
iCon.Execute iStr

iStr = "DELETE * FROM t_collate"
iCon.Execute iStr

iStr = "DELETE * FROM t_box_load"
iCon.Execute iStr

iStr = "SELECT * FROM t_box"
Rs.Open iStr, iCon, adOpenDynamic, adLockOptimistic

If Not (Rs.EOF And Rs.BOF) Then
Rs.MoveFirst
Do
Captor = GetFldData(Rs!box_quantity)
If (Captor >= 1) Then

For Counter = 1 To Captor
'Eliminate zero data
If (Rs!box_height > 0) And (Rs!box_width > 0) _
And (Rs!box_depth > 0) And (Rs!box_weight >= 0) Then

tempStr = GetNewTableID("t_box_load", AppsDataSource)
xStr = "INSERT INTO t_box_load " _
& "(t_box_load_id,t_box_id,box_width,box_height " _
& ",box_depth,box_weight,f_color_id,box_priority " _
& ",box_color,box_created_date) " _
& "VALUES ( " & tempStr & " , " & Rs!t_box_id & " " _
& ", " & GetFldData(Rs!box_height, 0) & " , " & GetFldData(Rs!box_width, 0) & " " _
& ", " & GetFldData(Rs!box_depth, 0) & " , " & GetFldData(Rs!box_weight, 0) & " " _
& ", " & GetFldData(Rs!f_box_id, "NULL") & " " _
& ", " & GetFldData(Rs!box_priority, 0) & " " _

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

& "," & GetFldData(Rs!box_color, 0) & ",#" & Now() & "# " _
& ")")
iCon.Execute xStr
Debug.Print xStr
End If
Next
End If
Rs.MoveNext
Loop Until Rs.EOF
End If
DoEvents
iCon.Close
Set iCon = Nothing

End Sub

Private Function getMinBoxDimension(ByRef Value As Single) As Boolean
Dim Rs           As New ADODB.Recordset
Dim iStr         As String
Dim ConnectStr   As String

ConnectStr = CreateMsAccessConnectionString(AppsDataSource)

'with one blow
iStr = "SELECT TOP 1 * FROM " _
& " (SELECT t_box_load_id ,t_box_id " _
& " ,IIF( (box_width <= box_depth ),'w','d' ) AS box_alignment " _
& " ,IIF( (box_width <= box_depth ),box_width ,box_depth ) AS min_value " _
& " FROM t_box_load " _
& " ) AS box_a " _
& "ORDER BY min_value"

'Debug.Print iStr

Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly

getMinBoxDimension = False

If Not (Rs.EOF And Rs.BOF) Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Value = GetFldData(Rs!min_value, 0)
't_box_id = GetFldData(Rs!t_box_id, 0)
'box_alignmenr = GetFldData(Rs!box_alignment, "")
getMinBoxDimension = True
End If

Rs.Close

Set Rs = Nothing

End Function

Private Function getMaxBoxDimension(ByRef Value As Single) As Boolean
Dim Rs As New ADODB.Recordset
Dim iStr As String
Dim ConnectStr As String

ConnectStr = CreateMsAccessConnectionString(AppsDataSource)

'with one blow
iStr = "SELECT TOP 1 * FROM " _
& " (SELECT t_box_load_id ,t_box_id " _
& " ,IIF( (box_width >= box_depth ),'w' , 'd' ) AS box_alignment " _
& " ,IIF( (box_width >= box_depth ),box_width ,box_depth ) AS max_value " _
& " FROM t_box_load " _
& " ) AS box_a " _
& "ORDER BY max_value"

'Debug.Print iStr
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly

getMaxBoxDimension = False
If Not (Rs.EOF And Rs.BOF) Then
Value = GetFldData(Rs!min_value, 0)
't_box_id = GetFldData(Rs!t_box_id, 0)
'box_alignmenr = GetFldData(Rs!box_alignment, "")
getMaxBoxDimension = True
End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Rs.Close

Set Rs = Nothing

End Function

Private Function getMaxBoxDimensionDx(ByRef Value As Single) As Boolean

Dim Rs As New ADODB.Recordset

Dim iStr As String

Dim ConnectStr As String

ConnectStr = CreateMsAccessConnectionString(AppsDataSource)

'with one blow

```
iStr = "SELECT TOP 1 * FROM " & _  
& " (SELECT t_box_load_id ,t_box_id " & _  
& " ,IIF( (box_width >= box_depth ),'w','d') AS box_alignment " & _  
& " ,IIF( (box_width >= box_depth ),box_width ,box_depth ) AS max_value " & _  
& " FROM t_box_load " & _  
& " ) AS box_a " & _  
& "ORDER BY max_value"
```

'Debug.Print iStr

Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly

getMaxBoxDimensionDx = False

If Not (Rs.EOF And Rs.BOF) Then

Value = GetFldData(Rs!min_value, 0)

't_box_id = GetFldData(Rs!t_box_id, 0)

'box_alignmenr = GetFldData(Rs!box_alignment, "")

getMaxBoxDimensionDx = True

End If

Rs.Close

Set Rs = Nothing

End Function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Function GetLoadingInformation(ByVal TLoadingID As String _
```

```
, Height As Single, Width As Single, Depth As Single _
```

```
, Optional WidthIsMax As Boolean = True) As Boolean
```

```
Dim Rs As New ADODB.Recordset
```

```
Dim iStr As String
```

```
Dim ConnectStr As String
```

```
iStr = "SELECT * FROM t_box_load " _
```

```
& "WHERE t_box_Load_id=" & TLoadingID & " "
```

```
ConnectStr = CreateMsAccessConnectionString(AppsDataSource)
```

```
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly
```

```
If Not (Rs.EOF And Rs.BOF) Then
```

```
GetLoadingInformation = True
```

```
Height = Rs!box_height
```

```
Width = Rs!box_width
```

```
Depth = Rs!box_depth
```

```
If (WidthIsMax) Then
```

```
Call SingleSwapWithMaxLeft(Width, Depth)
```

```
Else
```

```
Call SingleSwapWithMaxLeft(Depth, Width)
```

```
End If
```

```
Else
```

```
GetLoadingInformation = False
```

```
End If
```

```
Rs.Close
```

```
Set Rs = Nothing
```

```
End Function
```

```
Private Function getMaxBoxDimensionEx(ByRef Value As Single _
```

```
, ByRef tBoxLoadID As String) As Boolean
```

```
Dim Rs As New ADODB.Recordset
```

```
Dim iStr As String
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim ConnectStr As String
```

```
ConnectStr = CreateMsAccessConnectionString(AppsDataSource)
```

```
'with one blow
```

```
iStr = "SELECT TOP 1 * FROM " _
```

```
& " (SELECT t_box_load_id,t_box_id " _
```

```
& " ,IIF( (box_width >= box_depth ),'w','d' ) AS box_alignment " _
```

```
& " ,IIF( (box_width >= box_depth ),box_width,box_depth ) AS max_value " _
```

```
& " FROM t_box_load " _
```

```
& " ) AS box_a " _
```

```
& "ORDER BY max_value"
```

```
'Debug.Print iStr
```

```
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly
```

```
getMaxBoxDimensionEx = False
```

```
If Not (Rs.EOF And Rs.BOF) Then
```

```
Value = GetFldData(Rs!max_value, 0)
```

```
tBoxLoadID = Rs!t_box_load_id
```

```
getMaxBoxDimensionEx = True
```

```
End If
```

```
Rs.Close
```

```
Set Rs = Nothing
```

```
End Function
```

```
Private Function ComparBox()
```

```
End Function
```

```
Private Sub RemoveLoad(ByVal TLoadingID As String)
```

```
Dim iConn As New ADODB.Connection
```

```
Dim iStr As String
```

```
Dim ConnectStr As String
```

```

iStr = "DELETE * FROM t_box_load " _
& "WHERE t_box_Load_id=" & TLoadingID & " "
ConnectStr = CreateMsAccessConnectionString(AppsDataSource)
iConn.ConnectionString = ConnectStr
iConn.Open
iConn.Execute iStr
iConn.Close
Set iConn = Nothing

```

End Sub

Sub SolvedWithWallBuilding()

```

Dim bReachEndOfSolution As Boolean
Dim floatLayerDepth As Single
Dim t_box_id As Long
Dim strBoxAlignment As String
Dim bRet As Boolean

Dim AvailableDepth As Single
Dim layerDepth As Single
Dim bHaltLoop As Boolean
Dim sTemp As Single
Dim strBaseBoxId As String
Dim ContainerSpace As CoordBoxType
Dim Counter As Long

```

iCurrentLayer = 0

iConsumeSpace = 0

LoadWeight = 0

bReachEndOfSolution = False

iContainerWidth = Val(iForm.txContainer(0))

iContainerHeight = Val(iForm.txContainer(1))

iContainerDepth = Val(iForm.txContainer(2))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

iContainerMaxWeight = Val(iForm.txContainer(3))

ContainerSpace.Location.xx = 0

ContainerSpace.Location.yy = 0

ContainerSpace.Location.zz = 0

ContainerSpace.Width = iContainerWidth

ContainerSpace.Height = iContainerHeight

ContainerSpace.Depth = iContainerDepth

sTemp = iContainerWidth * iContainerHeight * iContainerDepth

iForm.txContainerVolume = Format(sTemp, "#,###.00")

AvailableDepth = iContainerDepth

Call PrepareDb4NewCalculation

layerDepth = 0

bHaltLoop = False

iOrder = 0

PinPoint.xx = -1

PinPoint.yy = -1

PinPoint.zz = -1

Do

layerDepth = ChooseWallDepth(ContainerSpace, strBaseBoxId)

If (layerDepth > 0) Then

If (ContainerSpace.Depth - iLayer(iCurrentLayer).Depth) > 0 Then

'heart of application

Call AFillStripe(iLayer(iCurrentLayer))

ContainerSpace.Location.zz = iLayer(iCurrentLayer).Depth _

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

+ iLayer(iCurrentLayer).Location.zz
ContainerSpace.Depth = ContainerSpace.Depth - iLayer(iCurrentLayer).Depth
Else
bHaltLoop = True
End If
Else
'meant not enough space to build wall
bHaltLoop = True
End If

Loop Until bHaltLoop

iConsumeSpace = 0
For Counter = 1 To iCurrentLayer
iConsumeSpace = iConsumeSpace + iLayer(Counter).dDepth
Next

iForm.txTotalLayer = Format(iCurrentLayer, "###0")
iForm.txUsageVolume = Format(iConsumeSpace, "###0")
iForm.txRemainVolume = Format(sTemp - iConsumeSpace, "###0")
iForm.txUsageVolumePercentage = Format((iConsumeSpace / sTemp) * 100, "##0.00")

Call InterfaceResultCollate(iForm.grdCollate)
Call BuildGraphicBoxCollate

End Sub
Function ChooseWallDepth(RemainContainerSpace As CoordType _
, baseBoxId As String) As Single

Dim tBoxLoadID As String
Dim iHeight As Single
Dim iWidth As Single
Dim iDepth As Single
Dim bChooseFlag As Boolean

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dim sTemp As Single
Dim BoxSpace As CoorBoxType

Dim sValue As Single

Dim MaxWeight As Single

MaxWeight = iContainerMaxWeight - LoadWeight

If Not FindLargestBoxWithSpace(RemainContainerSpace _
, tBoxLoadID, BoxSpace, MaxWeight) Then

ChooseWallDepth = -1

Exit Function

End If

ChooseWallDepth = BoxSpace.Depth

iCurrentLayer = iCurrentLayer + 1

ReDim Preserve iLayer(1 To iCurrentLayer) As CoorBoxType

iLayer(iCurrentLayer).Location.xx = 0

iLayer(iCurrentLayer).Location.yy = 0

If iCurrentLayer > 1 Then

iLayer(iCurrentLayer).Location.zz = iLayer(iCurrentLayer - 1).Location.zz _

+ iLayer(iCurrentLayer - 1).Depth

Else

iLayer(iCurrentLayer).Location.zz = 0

End If

iLayer(iCurrentLayer).Width = iContainerWidth

iLayer(iCurrentLayer).Height = iContainerHeight

iLayer(iCurrentLayer).Depth = BoxSpace.Depth

End Function

Private Function FindLargestBoxWithSpace(iSpace As CoorBoxType _

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

, BoxID As String, BoxSpace As CoordType, ByVal MaxWeight As Single _
, Optional FindAndPacked As Boolean = False) As Boolean

Dim Rs As New ADO.DB.Recordset

Dim iStr As String

Dim ConnectStr As String

Dim tBoxLoadID As String

Dim tBoxID As String

Dim iHeight As Single

Dim iWidth As Single

Dim iDepth As Single

Dim bChooseFlag As Boolean

Dim sValue As Single

If Not getMinBoxDimension(sValue) Then

FindLargestBoxWithSpace = False

Exit Function

End If

If (sValue > iSpace.Depth) _

Or (sValue > iSpace.Height) _

Or (sValue > iSpace.Width) Then

'not enough space

FindLargestBoxWithSpace = False

Exit Function

End If

ConnectStr = CreateMsAccessConnectionString(AppsDataSource)

'with one blow

iStr = "SELECT * FROM " _

& " (SELECT t_box_load_id,t_box_id,box_height,box_priority,box_color,box_weight " _

& " ,IIF((box_width >= box_depth),'w','d') AS box_alignment " _

& " ,IIF((box_width >= box_depth),box_width,box_depth) AS max_value " _

& " FROM t_box_load " _

& " WHERE box_weight <= " & MaxWeight & " " _

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

& " ) AS box_a " _
& "ORDER BY box_priority ASC, max_value DESC ,box_height DESC"
'insert priority
'Debug.Print iStr
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly
bChooseFlag = False
If Not (Rs.EOF And Rs.BOF) Then
Do
Call GetLoadingInformation(Rs!t_box_load_id, iHeight, iWidth, iDepth, True)
If (iHeight > iSpace.Height) Then
Rs.Close
Set Rs = Nothing
FindLargestBoxWithSpace = False
Exit Function
End If
'Rotate 2 axist
*****
'Rotate box to fit space

BoxSpace.Location.xx = iSpace.Location.xx
BoxSpace.Location.yy = iSpace.Location.yy
BoxSpace.Location.zz = iSpace.Location.zz
BoxSpace.Priority = GetFldData(Rs!box_priority)
BoxSpace.Color = GetFldData(Rs!box_color)
BoxSpace.Weight = GetFldData(Rs!box_weight)

If (iSpace.Width >= iWidth) And (iSpace.Depth >= iDepth) Then
bChooseFlag = True
BoxSpace.Depth = iDepth
BoxSpace.Width = iWidth
BoxSpace.Height = iHeight
BoxID = Rs!t_box_load_id
tBoxID = Rs!t_box_id
Exit Do
ElseIf (iSpace.Width >= iDepth) And (iSpace.Depth >= iWidth) Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bChooseFlag = True
```

```
BoxSpace.Depth = iWidth
```

```
BoxSpace.Width = iDepth
```

```
BoxSpace.Height = iHeight
```

```
BoxID = Rs!t_box_load_id
```

```
tBoxID = Rs!t_box_id
```

```
Exit Do
```

```
End If
```

```
*****
```

```
Rs.MoveNext
```

```
Loop Until Rs.EOF
```

```
End If
```

```
If (bChooseFlag And FindAndPacked) Then
```

```
Call SaveSolution(BoxID, tBoxID, BoxSpace)
```

```
End If
```

```
FindLargestBoxWithSpace = bChooseFlag
```

```
Rs.Close
```

```
Set Rs = Nothing
```

```
End Function
```

```
Sub SaveSolution(ByVal tBoxLoadID As String _  
, ByVal tBoxID As String, BoxSpace As CoordBoxType)
```

```
Dim iCon As New ADODB.Connection
```

```
Dim iStr As String
```

```
Dim Rs As New ADODB.Recordset
```

```
Dim tempStr As String
```

```
Dim Counter As Long
```

```
Dim Captor As Long
```

```
Dim xStr As String
```

```
'On Error Resume Next
```

```
iCon.ConnectionString = CreateMsAccessConnectionString(AppsDataSource)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

iCon.Open
iOrder = iOrder + 1
tempStr = GetNewTableID("t_collate", AppsDataSource)
iStr = "INSERT INTO t_collate " _
& "(t_collate_id,t_box_id ,collate_xx ,collate_yy ,collate_zz " _
& " ,width ,height ,depth ,weight ,priority ,color ,collate_sequenz )" _
& "VALUES " _
& "(" & tempStr & " , " & tBoxID & " " _
& " , " & BoxSpace.Location.xx & " , " & BoxSpace.Location.yy _
& " , " & BoxSpace.Location.zz _
& " , " & BoxSpace.Width & " , " & BoxSpace.Height _
& " , " & BoxSpace.Depth & " " _
& " , " & BoxSpace.Weight & " " _
& " , " & BoxSpace.Priority & " " _
& " , " & BoxSpace.Color & " , " & iOrder & ")"

```

```

iCon.Execute iStr
iCon.Close
Set iCon = Nothing
Call RemoveLoad(tBoxLoadID)

```

End Sub

```
Sub AFillWall(WallSpace As CoorBoxType)
```

```

Dim sValue As Single
Dim tBoxLoadID As String
Dim iHeight As Single
Dim iWidth As Single
Dim iDepth As Single
Dim bChooseFlag As Boolean
Dim sTemp As Single
Dim BoxSpace As CoorBoxType
Dim StripeSpace As CoorBoxType
Dim KeepSpace As CoorBoxType

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Static MaxWidth As Single

Static MaxHeight As Single

Static CapLayer As Integer

Dim temps As Single

Dim zWidth As Single

Dim zHeight As Single

Dim MaxWeight As Single

'create hstripe

LSet StripeSpace = WallSpace

LSet KeepSpace = WallSpace

Do

**If FindLargestBoxWithSpace(KeepSpace _
, tBoxLoadID, BoxSpace, MaxWeight, False) Then**

'Kept ref box n make horizon stripe

StripeSpace.Height = BoxSpace.Height

Call AFillStripe(StripeSpace)

StripeSpace.Location.yy = StripeSpace.Location.yy + BoxSpace.Height

End If

Loop Until (WallSpace.Height >= StripeSpace.Location.yy)

End Sub

Sub AFillStripe(WallSpace As CoorBoxType)

Dim sValue As Single

Dim tBoxLoadID As String

Dim iHeight As Single

Dim iWidth As Single

Dim iDepth As Single

Dim bChooseFlag As Boolean

Dim sTemp As Single

Dim BoxSpace As CoorBoxType

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Dim StripeSpace As CoorBoxType

Dim KeepSpace As CoorBoxType

Static MaxWidth As Single

Static MaxHeight As Single

Static CapLayer As Integer

Dim temps As Single

Dim zWidth As Single

Dim zHeight As Single

Dim MaxWeight As Single

MaxWeight = iContainerMaxWeight - LoadWeight

If Not FindLargestBoxWithSpace(WallSpace _

, tBoxLoadID, BoxSpace, MaxWeight, True) Then

'iConsumeSpace = iConsumeSpace _

' + (MaxHeight * MaxWidth * iLayer(CapLayer).Depth)

Exit Sub

End If

'Simple Weight

LoadWeight = LoadWeight + BoxSpace.Weight

If (LoadWeight > iContainerMaxWeight) Then

Exit Sub

End If

'Store layer dimension

zHeight = BoxSpace.Location.yy + BoxSpace.Height

zWidth = BoxSpace.Location.xx + BoxSpace.Width

If (MaxHeight < zHeight) Then

MaxHeight = zHeight

iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _

* iLayer(iCurrentLayer).Depth)

End If

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If (MaxWidth < zWidth) Then
MaxWidth = zWidth
iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _
* iLayer(iCurrentLayer).Depth)
End If

```

```

If (CapLayer <> iCurrentLayer) Then
CapLayer = iCurrentLayer
MaxHeight = 0
MaxWidth = 0
End If

```

```

'Find stripe heightdefination
If (WallSpace.Location.xx = 0) _
And (WallSpace.Location.yy <> PinPoint.yy) Then
PinPoint.yy = WallSpace.Location.yy
End If

```

```

If (WallSpace.Depth - BoxSpace.Depth) > 0 Then
LSet StripeSpace = WallSpace
StripeSpace.Depth = WallSpace.Depth - BoxSpace.Depth
StripeSpace.Width = BoxSpace.Width
StripeSpace.Height = BoxSpace.Height
StripeSpace.Location.zz = WallSpace.Location.zz + BoxSpace.Depth
Call AFillStripe(StripeSpace)
End If

```

```

If (WallSpace.Width - BoxSpace.Width) > 0 Then
LSet StripeSpace = WallSpace
StripeSpace.Width = WallSpace.Width - BoxSpace.Width
StripeSpace.Location.xx = WallSpace.Location.xx + BoxSpace.Width
Call AFillStripe(StripeSpace)
End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If (WallSpace.Height - BoxSpace.Height) > 0 Then
LSet StripeSpace = WallSpace
StripeSpace.Height = WallSpace.Height - BoxSpace.Height
StripeSpace.Width = BoxSpace.Width
StripeSpace.Location.yy = WallSpace.Location.yy + BoxSpace.Height
Call AFillStripe(StripeSpace)
End If

```

End Sub

```

Sub BFillStripe(WallSpace As CoorBoxType)

```

```

Dim sValue As Single

```

```

Dim tBoxLoadID As String

```

```

Dim iHeight As Single

```

```

Dim iWidth As Single

```

```

Dim iDepth As Single

```

```

Dim bChooseFlag As Boolean

```

```

Dim sTemp As Single

```

```

Dim BoxSpace As CoorBoxType

```

```

Dim StripeSpace As CoorBoxType

```

```

Dim KeepSpace As CoorBoxType

```

```

Static MaxWidth As Single

```

```

Static MaxHeight As Single

```

```

Static CapLayer As Integer

```

```

Dim temps As Single

```

```

Dim zWidth As Single

```

```

Dim zHeight As Single

```

```

Dim MaxWeight As Single

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MaxWeight = iContainerMaxWeight - LoadWeight
If Not FindLargestBoxWithSpace(WallSpace _
, iBoxLoadID, BoxSpace, MaxWeight, True) Then
'iConsumeSpace = iConsumeSpace _
' + (MaxHeight * MaxWidth * iLayer(CapLayer).Depth)
Exit Sub
End If

```

```

'Simple Weight
LoadWeight = LoadWeight + BoxSpace.Weight
If (LoadWeight > iContainerMaxWeight) Then
Exit Sub
End If

```

```

'Store layer dimension
zHeight = BoxSpace.Location.yy + BoxSpace.Height
zWidth = BoxSpace.Location.xx + BoxSpace.Width

```

```

If (MaxHeight < zHeight) Then
MaxHeight = zHeight
iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _
* iLayer(iCurrentLayer).Depth)
End If

```

```

If (MaxWidth < zWidth) Then
MaxWidth = zWidth
iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _
* iLayer(iCurrentLayer).Depth)
End If

```

```

If (CapLayer <> iCurrentLayer) Then
CapLayer = iCurrentLayer
MaxHeight = 0
MaxWidth = 0
End If

```

'Find stripe heightdefination

If (WallSpace.Location.xx = 0) _

And (WallSpace.Location.yy <> PinPoint.yy) Then

PinPoint.yy = WallSpace.Location.yy

End If

'create hstripe

Do

StripeSpace.Location.zz = WallSpace.Location.zz

'cap first box height

StripeHorizonHeight = BoxSpace.Height

'similar wall creation

Do

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.yy = WallSpace.Location.yy

StripeSpace.Depth = BoxSpace.Depth

StripeSpace.Height = StripeHorizonHeight

StripeSpace.Width = WallSpace.Width - BoxSpace.Width

'recursive fill sub horizon stripe

Call AFillWall(StripeSpace)

StripeSpace.Location.zz = StripeSpace.Location.zz + BoxSpace.Depth

Loop Until StripeSpace.Location.zz > (WallSpace.Location.zz + WallSpace.Depth)

'meant completed fill horizon stripe and reduce wallspace height

WallSpace.Height = WallSpace.Height - BoxSpace.Height

WallSpace.Location.yy = WallSpace.Location.yy + BoxSpace.Height

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.yy = WallSpace.Location.yy + StripeHorizonHeight

StripeSpace.Location.zz = WallSpace.Location.zz

StripeSpace.Depth = WallSpace.Depth

StripeSpace.Height = WallSpace.Height

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

StripeSpace.Width = WallSpace.Width

Call AFillWall(StripeSpace)

Loop Until (WallSpace.Height <= 0)

'Find Depth stripe

If (WallSpace.Depth - BoxSpace.Depth) > 0 Then

Do

StripeSpace.Location.zz = WallSpace.Location.zz

'cap first box height

StripeHorizonHeight = BoxSpace.Height

'similar wall creation

Do

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.yy = WallSpace.Location.yy

StripeSpace.Depth = BoxSpace.Depth

StripeSpace.Height = StripeHorizonHeight

StripeSpace.Width = WallSpace.Width · BoxSpace.Width

'recursive fill sub horizon stripe

Call AFillWall(StripeSpace)

StripeSpace.Location.zz = StripeSpace.Location.zz + BoxSpace.Depth

Loop Until StripeSpace.Location.zz > (WallSpace.Location.zz + WallSpace.Depth)

'meant completed fill horizon stripe and reduce wallspace height

WallSpace.Height = WallSpace.Height · BoxSpace.Height

WallSpace.Location.yy = WallSpace.Location.yy + BoxSpace.Height

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.yy = WallSpace.Location.yy + StripeHorizonHeight

StripeSpace.Location.zz = WallSpace.Location.zz

StripeSpace.Depth = WallSpace.Depth

StripeSpace.Height = WallSpace.Height

StripeSpace.Width = WallSpace.Width

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Call AFillWall(StripeSpace)

Loop Until (WallSpace.Height <= 0)

End If

'implement filling fncion

'associated fill horizon stripe

If (WallSpace.Width - BoxSpace.Width) > 0 Then

StripeSpace.Width = WallSpace.Width - BoxSpace.Width

Do While (StripeSpace.Weight > 0)

StripeSpace.Location.xx = WallSpace.Location.xx + BoxSpace.Width

StripeSpace.Location.yy = WallSpace.Location.yy

StripeSpace.Location.zz = WallSpace.Location.zz

StripeSpace.Depth = WallSpace.Depth

StripeSpace.Height = WallSpace.Height

Call AFillWall(StripeSpace)

StripeSpace.Width = StripeSpace.Width - BoxSpace.Width

Loop

Else

End If

'Find Horizon stripe -> reset to next row

If (WallSpace.Height - BoxSpace.Height) > 0 Then

StripeSpace.Height = WallSpace.Height - BoxSpace.Height

StripeSpace.Location.yy = WallSpace.Location.yy + BoxSpace.Height

Do While (StripeSpace.Height > 0)

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.zz = WallSpace.Location.zz

StripeSpace.Depth = WallSpace.Depth

StripeSpace.Width = WallSpace.Width

'StripeSpace.Height = StripeHorizonHeight

```

Call AFillWall(StripeSpace)
StripeSpace.Location.yy = StripeSpace.Location.yy + BoxSpace.Height
StripeSpace.Height = StripeSpace.Height - BoxSpace.Height
Loop
Else
Debug.Print "Jugkuuu"
End If

End Sub

```

```

Sub BFillWall(WallSpace As CoorBoxType)

```

```

Dim sValue As Single
Dim tBoxLoadID As String
Dim iHeight As Single
Dim iWidth As Single
Dim iDepth As Single
Dim bChooseFlag As Boolean
Dim sTemp As Single
Dim BoxSpace As CoorBoxType
Dim StripeSpace As CoorBoxType
Dim KeepSpace As CoorBoxType

```

```

Static MaxWidth As Single
Static MaxHeight As Single
Static CapLayer As Integer

```

```

Static StripeHorizonHeight As Single

```

```

Dim temps As Single
Dim zWidth As Single
Dim zHeight As Single
Dim MaxWeight As Single

```

```

MaxWeight = iContainerMaxWeight - LoadWeight
If Not FindLargestBoxWithSpace(WallSpace _
, tBoxLoadID, BoxSpace, MaxWeight, True) Then
'iConsumeSpace = iConsumeSpace _
' + (MaxHeight * MaxWidth * iLayer(CapLayer).Depth)
Exit Sub
End If

```

```

'Simple Weight
LoadWeight = LoadWeight + BoxSpace.Weight
If (LoadWeight > iContainerMaxWeight) Then
Exit Sub
End If

```

```

'Store layer dimension
zHeight = BoxSpace.Location.yy + BoxSpace.Height
zWidth = BoxSpace.Location.xx + BoxSpace.Width

If (MaxHeight < zHeight) Then
MaxHeight = zHeight
iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _
* iLayer(iCurrentLayer).Depth)
End If

```

```

If (MaxWidth < zWidth) Then
MaxWidth = zWidth
iLayer(iCurrentLayer).dDepth = (MaxHeight * MaxWidth _
* iLayer(iCurrentLayer).Depth)
End If

```

```

If (CapLayer <> iCurrentLayer) Then
CapLayer = iCurrentLayer
MaxHeight = 0
MaxWidth = 0
End If

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'Find stripe heightdefination

If (WallSpace.Location.xx = 0) _

And (WallSpace.Location.yy <> PinPoint.yy) Then

PinPoint.yy = WallSpace.Location.yy

End If

'Find Depth stripe

If (WallSpace.Depth - BoxSpace.Depth) > 0 Then

StripeSpace.Location.xx = WallSpace.Location.xx

StripeSpace.Location.yy = WallSpace.Location.yy

StripeSpace.Location.zz = WallSpace.Location.zz + BoxSpace.Depth

StripeSpace.Depth = WallSpace.Depth - BoxSpace.Depth

StripeSpace.Height = WallSpace.Height

StripeSpace.Width = BoxSpace.Width

Call AFillWall(StripeSpace)

Else

End If

'Find Horizon stripe -> same row

If (WallSpace.Width - BoxSpace.Width) > 0 Then

StripeSpace.Location.xx = WallSpace.Location.xx + BoxSpace.Width

StripeSpace.Location.yy = WallSpace.Location.yy

StripeSpace.Location.zz = WallSpace.Location.zz

StripeSpace.Depth = WallSpace.Depth

StripeSpace.Height = WallSpace.Height

StripeSpace.Width = WallSpace.Width - BoxSpace.Width

If (BoxSpace.Height > StripeHorizonHeight) Then

StripeHorizonHeight = BoxSpace.Height

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
StripeSpace.Height = StripeHorizonHeight
Call AFillWall(StripeSpace)
Else
StripeHorizonHeight = 0
End If

```

```

'Find Horizon stripe -> reset to next row
If (WallSpace.Height - BoxSpace.Height) > 0 Then

```

```

StripeSpace.Location.xx = WallSpace.Location.xx
StripeSpace.Location.yy = WallSpace.Location.yy + BoxSpace.Height
StripeSpace.Location.zz = WallSpace.Location.zz

```

```

StripeSpace.Depth = WallSpace.Depth
StripeSpace.Width = WallSpace.Width
StripeSpace.Height = WallSpace.Height - BoxSpace.Height
'StripeSpace.Height = StripeHorizonHeight

```

```

Call AFillWall(StripeSpace)
Else
Debug.Print "Jugkuuu"
End If

```

```

End Sub

```

```

Sub InterfaceResultColatte(uGrid As SGGrid)

```

```

Dim Rs As New Recordset
Dim iStr As String
Dim ConnectStr As String
Dim tBoxLoadID As String

```

```

Dim iHeight As Single
Dim iWidth As Single

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Dim iDepth As Single
Dim bChooseFlag As Boolean
Dim iCol As SGColumn

```

```

uGrid.DataMode = sgUnbound 'Reset
uGrid.DataMode = sgBound 'Reset

```

```

ConnectStr = CreateMsAccessConnectionString(AppsDataSource)

```

```

'with one blow
iStr = "SELECT t_collate.collate_sequenz as series ,box_alias " _
& ",t_collate.* ,box_comment FROM t_collate ,t_box " _
& "WHERE (t_collate.t_box_id=t_box.t_box_id)" _
& "ORDER BY t_collate.t_collate_id"

```

```

'Debug.Print iStr
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly
bChooseFlag = False
Set uGrid.DataSource = Rs

```

```

With uGrid
.ReBind
.EnterKeyBehavior = sgEnterKeyColumns

```

```

For Each iCol In .Columns
iCol.ReadOnly = True
Select Case iCol.DBField
Case "series"
iCol.Caption = "ลำดับ"
iCol.Width = 800
iCol.Style.TextAlign = sgAlignCenterCenter
Case "box_alias"
iCol.Caption = "ชื่อ"
iCol.Width = 2000
Case "width"
iCol.Caption = "ความกว้าง"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

iCol.Width = 1000

Case "height"

iCol.Caption = "È'èÒÈÛ§"

iCol.Width = 1000

Case "depth"

iCol.Caption = "๐ÇÒÁÁÖ;"

iCol.Width = 1000

Case "collate_xx"

iCol.Caption = "x"

iCol.Width = 1000

Case "collate_yy"

iCol.Caption = "y"

iCol.Width = 1000

Case "collate_zz"

iCol.Caption = "z"

iCol.Width = 1200

Case "priority"

iCol.Caption = "ÁÓ'Ñ°;ÒÃ'Ñ'Êes"

iCol.Width = 1400

iCol.AllowSizing = False

gbCol_Loads_FrameColor = iCol.ColIndex

Case "box_comment"

iCol.Caption = "ÈÁÒÁáÈµØ"

iCol.Width = 2000

Case Else

iCol.Hidden = True

iCol.AllowSizing = False

End Select

Next

End With

End Sub

Public Sub BuildGraphicBoxCollate(Optional isAnimated As Boolean = False)

Dim Rs As New Recordset

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim iStr As String
Dim ConnectStr As String
Dim igBox As cGBox
```

```
Dim CxCounter As Long
```

```
Dim AngelX
```

```
Dim AngelY
```

```
Dim AngelZ
```

```
AngelX = 350
```

```
AngelY = 30
```

```
AngelZ = 10
```

```
ConnectStr = CreateMsAccessConnectionString(AppsDataSource)
```

```
iStr = "SELECT t_collate.collate_sequenz as series ,box_alias " _  
& ",t_collate.* ,box_comment FROM t_collate ,t_box " _  
& "WHERE (t_collate.t_box_id=t_box.t_box_id)" _  
& "ORDER BY t_collate.t_collate_id ASC"
```

```
Rs.Open iStr, ConnectStr, adOpenForwardOnly, adLockReadOnly
```

```
iForm.iPicLayout.Cls
```

```
Screen.MousePointer = vbHourglass
```

```
DoEvents
```

```
If Not (Rs.EOF And Rs.BOF) Then
```

```
Do
```

```
Set igBox = New cGBox
```

```
With igBox
```

```
Set .PicControl = iForm.iPicLayout
```

```
.Angel_x = AngelX
```

```
.Angel_y = AngelY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
.Angel_z = AngelZ
```

```
.Point_xx = Rs!collate_xx
```

```
.Point_yy = Rs!collate_yy
```

```
.Point_zz = Rs!collate_zz
```

```
.Dimension_Width = Rs!Width
```

```
.Dimension_Height = Rs!Height
```

```
.Dimension_Depth = Rs!Depth
```

```
.LineColor = Rs!Color
```

```
.ZoomFactor = 10
```

```
.Draw
```

```
End With
```

```
Set igBox = Nothing
```

```
Rs.MoveNext
```

```
If isAnimated Then
```

```
For CxCounter = 1 To 4000000
```

```
Next
```

```
End If
```

```
Loop Until Rs.EOF
```

```
End If
```

```
Rs.Close
```

```
Set Rs = Nothing
```

```
'draw container
```

```
Set igBox = New cGBox
```

```
With igBox
```

```
Set .PicControl = iForm.iPicLayout
```

```
.Angel_x = AngelX
```

```
.Angel_y = AngelY
```

```
.Angel_z = AngelZ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
.Point_xx = 0
.Point_yy = 0
.Point_zz = 0

.Dimension_Width = iContainerWidth
.Dimension_Height = iContainerHeight
.Dimension_Depth = iContainerDepth
```

```
.LineColor = vbBlue
.ZoomFactor = 10
.Draw True
```

```
End With
```

```
Set igBox = Nothing
```

```
Screen.MousePointer = vbDefault
```

```
DoEvents
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

ข้อมูลขนาดของคอนเทนเนอร์

ตารางที่ ผค 1 ข้อมูลคอนเทนเนอร์ขนาดต่างๆ

Description		20-Foot		40-Foot		40-Foot High Cube		45-Foot
		Aluminium	steel	Aluminium	steel	Aluminium	steel	Aluminium
External Dimensions	Length (mm.)	6,058	6,058	12,192	12,192	12,192	12,192	13,716
	Width (mm.)	2,438	2,438	2,438	2,438	2,438	2,438	2,438
	Height (mm.)	2,591	2,591	2,591	2,591	2,896	2,896	2,896
Inter Dimensions	Length (mm.)	5,926	5,899	12,052	12,033	12,052	12,003	13,576
	Width (mm.)	2,349	2,352	2,346	2,352	2,346	2,352	2,345
	Height (mm.)	2,382	2,386	2,382	2,368	2,687	2,691	2,686
Capacity	Cubic Measurement (M ³)	33.2	33.1	67.3	67.5	79.5	76.2	85.5
	Tare Weight (kg)	1,790	2,220	2,870	3,740	3,000	3,920	2,343
	Payload (kg)	22,210	21,780	27,610	26,740	27,480	26,560	2,585
	Gross Weight (kg)	24,000	24,000	30,480	30,480	30,480	30,480	3,950
Door Opening	Width (mm.)	2,343	2,340	2,343	2,340	2,343	2,340	26,530
	Height (mm.)	2,297	2,272	2,279	2,272	2,585	2,577	30,480

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้