



ภาคผนวก ก

มหาวิทยาลัยราชภัฏสุรินทร์

ในส่วนนี้จะแสดงถึง

1. ตารางปัญหาที่ 1 – 6 จากวิทยานิพนธ์ของดร.ขวัญนิตี คำเมือง
2. ตารางผลลัพธ์ของปัญหา 1 – 6 ในกรณีที่ 1 คือ การเปรียบเทียบจำนวนหน่วยการผลิต, จำนวนของ Exceptional Elements และจำนวนของ Voids ในกรณีที่ต้องการให้จำนวนของ Exceptional Elements น้อยที่สุดและมีจำนวนหน่วยการผลิตเท่ากับผลลัพธ์หน่วยการผลิตของ GA โดยใช้โปรแกรม ROC1 และ ROC2 ในการหาผลลัพธ์
3. ตารางผลลัพธ์ของปัญหา 1 – 6 ในกรณีที่ 2 คือ การเปรียบเทียบจำนวนหน่วยการผลิต, จำนวนของ Exceptional Elements และจำนวนของ Voids ในกรณีที่ต้องการให้ที่ต้องการให้จำนวนของ Voids น้อยที่สุด โดยใช้โปรแกรม ROC1 และ ROC2 ในการหาผลลัพธ์ เพื่อเปรียบเทียบผลลัพธ์จากโปรแกรม GA

ตารางที่ ก - 1 ถึง ก - 6 แสดงปัญหาที่ 1 – 6 จากวิทยานิพนธ์ของดร.ขวัญนิตี คำเมือง

ตารางที่ ก - 6 ถึง ก - 12 แสดงผลลัพธ์ของปัญหาในกรณีที่ 1

ตารางที่ ก - 13 ถึง ก - 18 แสดงผลลัพธ์ของปัญหาในกรณี 2



ตารางที่ ก - 2 ปัญหาที่ 2

		Parts									
		1	2	3	4	5	6	7	8	9	0
Machines	1	1	1				1				
	2	1									
	3	1					1				
	4			1				1	1		
	5			1					1		
	6				1						1
	7				1	1				1	1
	8				1	1				1	1
	9	1	1	1			1	1		1	
	10	1	1	1	1		1		1		

ตารางที่ ก - 3 ปัญหาที่ 3

		Parts					
		1	2	3	4	5	6
Machines	1		1		1		1
	2					1	1
	3			1		1	1
	4	1			1		
	5	1			1		
	6					1	1

ตารางที่ ก - 4 ปัญหาที่ 4

		Parts									
		1	2	3	4	5	6	7	8	9	0
Machines	1						1				
	2	1									
	3	1		1						1	
	4	1		1	1					1	
	5		1			1			1		1
	6					1	1	1			1
	7		1			1	1		1		
	8		1			1	1		1		1

ตารางที่ ก-5 ปัญหาที่ 5

		Parts									
		1	2	3	4	5	6	7	8	9	0
Machines	1						1	1			
	2	1									
	3	1		1						1	
	4	1		1	1					1	
	5		1			1			1		1
	6					1	1	1			
	7		1			1	1		1		
	8		1			1	1		1		1

ตารางที่ ก-6 ปัญหาที่ 6

		Parts															
		1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	
Machines	1			1	1	1		1			1	1	1	1	1	1	
	2	1	1											1			
	3									1					1		
	4		1								1		1	1			
	5			1	1											1	
	6				1		1									1	
	7			1			1									1	
	8					1		1				1					
	9							1				1					
	10					1		1				1				1	
	11							1				1				1	
	12								1						1		
	13									1	1				1		
	14	1	1														
	15								1	1					1		

ตารางที่ ก - 8 ผลลัพธ์ของปัญหาที่ 2 (ในกรณีที่ 1)

		Parts									
		1	6	2	3	8	4	7	9	5	0
Machines	10	1	1	1	1	1	1				
	9	1	1	1	1			1	1		
	1	1	1	1							
	3	1	1								
	2	1									
	4				1	1		1			
	5				1	1					
	7						1		1	1	1
	8						1		1	1	1
	6						1		1		

ตารางที่ ก - 9 ผลลัพธ์ของปัญหาที่ 3 (ในกรณีที่ 1)

		Parts					
		1	3	6	2	4	5
Machines	4	1	1				
	5	1	1				
	1			1	1	1	
	3			1	1		1
	2			1			1
	6					1	1

ตารางที่ ก - 10 ผลลัพธ์ของปัญหาที่ 4 (ในกรณีที่ 1)

		Parts									
		1	3	9	4	5	2	8	6	0	7
Machines	4	1	1	1	1						
	3	1	1	1							
	2	1									
	8					1	1	1	1	1	
	7					1	1	1	1		
	5					1	1	1		1	
	6					1			1	1	1
	1								1		

ตารางที่ ก - 11 ผลลัพธ์ของปัญหาที่ 5 (ในกรณีที่ 1)

		Parts									
		1	3	9	4	5	2	8	6	0	7
Machines	4	1	1	1	1						
	3	1	1	1							
	2	1									
	8					1	1	1	1	1	
	7					1	1	1	1		
	5					1	1	1		1	
	6					1			1		1
	1								1		1

ตารางที่ ก - 12 ผลลัพธ์ของปัญหาที่ 6 (ในกรณีที่ 1)

		Parts												
		2	1	1	1	1	1	1	1	1	1	1	1	1
Machines	2	1	1	1										
	14	1	1											
	4	1	1	1	1									
	1		1	1	1	1	1	1	1	1	1	1		
	13			1									1	1
	15			1									1	1
	12			1									1	
	3			1										1
	5					1	1	1						
	7					1	1							1
	6					1	1							1
	8								1	1	1			
	10								1	1	1			
9								1	1		1			
11								1	1		1			

ตารางที่ ก - 14 ผลลัพธ์ของปัญหาที่ 2 (ในกรณีที่ 2)

		Parts									
		1	6	2	3	8	4	7	9	5	0
Machines	10	1	1	1	1	1	1				
	9	1	1	1	1			1	1		
	1	1	1	1							
	3	1	1								
	2	1									
	4				1	1		1			
	5				1	1					
	7							1	1	1	1
	8							1	1	1	1
	6							1	1		

ตารางที่ ก - 15 ผลลัพธ์ของปัญหาที่ 3 (ในกรณีที่ 2)

		Parts					
		1	3	6	2	4	5
Machines	4	1	1				
	5	1	1				
	1			1	1	1	
	3			1	1		1
	2			1			1
	6					1	1

ตารางที่ ก - 16 ผลลัพธ์ของปัญหาที่ 4 (ในกรณีที่ 2)

		Part									
		1	3	9	4	5	2	8	6	0	7
Machines	4	1	1	1	1						
	3	1	1	1							
	2	1									
	8				1	1	1	1	1		1
	7				1	1	1	1			
	5				1	1	1			1	
	6				1				1	1	1
	1								1		

ตารางที่ ก - 17 ผลลัพธ์ของปัญหาที่ 5 (ในกรณีที่ 2)

		Parts									
		1	3	9	4	5	2	8	6	0	7
Machines	4	1	1	1	1						
	3	1	1	1							
	2	1									
	8					1	1	1	1	1	
	7					1	1	1	1		
	5					1	1	1		1	
	6					1			1		1
	1								1		1

ตารางที่ ก - 18 ผลลัพธ์ของปัญหาที่ 6 (ในกรณีที่ 2)

		Parts														
		2	1	2	3	0	4	3	4	7	1	5	5	8	9	6
Machines	2	1	1	1												
	14	1	1													
	4	1		1	1	1										
	1		1	1	1	1	1	1	1	1	1	1	1			
	13			1										1	1	
	15			1										1	1	
	12			1										1		
	3			1											1	
	5					1	1	1								
	7					1	1									1
	6					1		1								1
	8									1	1	1				
	10									1	1	1				
9									1	1		1				
11									1	1		1				



ในส่วนนี้จะกล่าวถึงโค้ดที่ใช้สร้างโปรแกรม และคุณสมบัติของปุ่มต่าง ๆ ในโปรแกรม ROC1 และ ROC2

ในส่วนของโค้ดของโปรแกรม ROC1 และ ROC2 จะมีความคล้ายคลึงกันมาก แต่จะแตกต่างกันในตอนที่ทำให้การคำนวณที่ตำแหน่ง Button2 ซึ่งจะมีวิธีการและโค้ดที่แตกต่างกัน

คุณสมบัติของ Controls ที่นำมาใช้ออกแบบ และสร้างโปรแกรม ROC1 และ ROC2

ตารางที่ ๗ - 1 คุณสมบัติของ Control

Common Controls	คุณสมบัติ	ค่าที่กำหนด
Label1	Name	Label1
	Text	Machines :
Label2	Name	Label2
	Text	Parts :
Label3	Name	Label3
	Text	Parts
Label4	Name	Label4
	Text	Machines
TextBox	Name	TextBox1
	Text	
TextBox	Name	TextBox2
	Text	
Button	Name	Button1
	Text	สร้างเมตริกซ์
Button	Name	Button2
	Text	แสดงผลลัพธ์
Button	Name	Button4
	Text	แสดง Number of Exceptional Elements และ Number of Voids

ตารางที่ ๑ - 1 คุณสมบัติของ Control (ต่อ)

Containers	คุณสมบัติ	ค่าที่กำหนด
GroupBox1	Name	GroupBox1
	Text	
Dialog	คุณสมบัติ	ค่าที่กำหนด
Open File Dialog	Name	OpenToolStripMenuItem
	Text	Open
Save File Dialog	Name	SaveToolStripMenuItem
	Text	Save

โค้ดของการสร้างเมตริกซ์ที่ Button1 ในโปรแกรม ROC1 และ ROC2

ตารางที่ ๑ - 2 โค้ดของการสร้างเมตริกซ์ที่ Button1 ในโปรแกรม ROC1 และ ROC2

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
If Me.TextBox1.Text = "" Or Me.TextBox2.Text = "" Then Exit Sub
chkChanged = False
chkInput = True
Me.GroupBox1.Controls.Clear()
Dim i As Integer, j As Integer
For i = 1 To Me.TextBox1.Text
Dim lbl As New Label
lbl.Name = "lbl0" & i
lbl.Tag = i & ",0"
lbl.BackColor = Color.Khaki
lbl.Size = New System.Drawing.Size(20, 20)
lbl.Location = New System.Drawing.Point(30, i * 30)
lbl.TextAlign = HorizontalAlignment.Center

```

ตารางที่ ๑ – 2 ได้ดของการสร้างเมตริกซ์ที่ Button1 ในโปรแกรม ROC1 และ ROC (ต่อ)

```
Me.GroupBox1.Controls.Add(lbl)
Dim txt1 As New TextBox
txt1.Name = "text" & i & "0"
txt1.BackColor = Color.Red
txt1.Tag = i & ",0"
txt1.Size = New System.Drawing.Size(25, 20)
txt1.Location = New System.Drawing.Point(Int(TextBox2.Text + 2) * 30, i * 30)
txt1.TextAlign = HorizontalAlignment.Center
txt1.Text = i
txt1.Visible = False
Me.GroupBox1.Controls.Add(txt1)
For j = 1 To Me.TextBox2.Text
If i = 1 Then
Dim lbl2 As New Label
lbl2.Name = "lbl" & j & "0"
lbl2.Tag = "0," & j
lbl2.BackColor = Color.Honeydew
lbl2.Size = New System.Drawing.Size(20, 20)
lbl2.Location = New System.Drawing.Point((j * 30) + 30, 0)
lbl2.TextAlign = HorizontalAlignment.Center
lbl2.Text = j
Me.GroupBox1.Controls.Add(lbl2)
Dim txt2 As New TextBox
txt2.Name = "text0" & j
txt2.Tag = "0," & j
txt2.BackColor = Color.Red
txt2.Size = New System.Drawing.Size(25, 20)
txt2.Location = New System.Drawing.Point((j * 30) + 30, Int(TextBox1.Text + 1) * 30)
txt2.TextAlign = HorizontalAlignment.Center
```

ตารางที่ ข – 2 โค้ดของการสร้างเมตริกซ์ที่ Button1 ในโปรแกรม ROC1 และ ROC (ต่อ)

```

txt2.Text = j
txt2.Visible = False
Me.GroupBox1.Controls.Add(txt2)
End If
Dim txt As New TextBox
txt.Name = "txt" & i & j
txt.Tag = i & "," & j
txt.MaxLength = 1
txt.Size = New System.Drawing.Size(25, 20)
txt.TextAlign = HorizontalAlignment.Center
txt.Location = New System.Drawing.Point((j * 30) + 30, i * 30)
Me.GroupBox1.Controls.Add(txt)
Me.GroupBox1.Size = New System.Drawing.Size((30 * Me.TextBox2.Text) + 100, (30 *
Me.TextBox1.Text) + 70)
Next
Next
End Sub

```

โค้ดของการคำนวณหาผลลัพธ์ที่ Button2 ในโปรแกรม ROC1 ประกอบไปด้วยฟังก์ชันย่อยดังต่อไปนี้

โค้ดการอ่านค่าในเมตริกซ์เพื่อนำมาเป็นไว้ใน Array 2 มิติ

ตารางที่ ข - 3 โค้ดการอ่านค่าในเมตริกซ์เพื่อนำมาเป็นไว้ใน Array 2 มิติในโปรแกรม ROC1

```

Private Sub ReadMatrix()
Dim x As Integer = Me.TextBox1.Text
Dim y As Integer = Me.TextBox2.Text
Dim i As Integer, j As Integer
Dim name(1) As String

```


ตารางที่ ข - 3 ได้จัดการอ่านค่าในเมตริกซ์เพื่อนำมาเป็นไว้ใน Array 2 มิติในโปรแกรม ROC1 (ต่อ)

```
Dim x1 As Integer, y1 As Integer
ReDim roc(x, y)
ReDim NOX(x)
ReDim NOY(y)
.For Each c As Control In Me.GroupBox1.Controls
If TypeOf c Is TextBox Then
Dim t As TextBox = CType(c, TextBox)
name = Split(t.Tag.ToString, ",")
x1 = name(0)
y1 = name(1)
roc(x1, y1) = If(t.Text = "", 0, t.Text)
End If
If TypeOf c Is Label Then
If c.Name.StartsWith("lbl") Then
Dim l As Label = CType(c, Label)
name = Split(l.Tag.ToString, ",")
x1 = name(0)
y1 = name(1)
If x1 = 0 Then
NOY(y1) = l.Text
Else
NOX(x1) = l.Text
End If
End If
End If
Next
```

โค้ดการหาค่าผลรวมในแต่ละแถว และคอลัมน์

ตารางที่ ข – 4 โค้ดการหาค่าผลรวมในแต่ละแถว และคอลัมน์

```

Dim sum As Double
For i = 1 To x
For j = 1 To y
sum += roc(i, j) * (2 ^ (y - j))
Next
roc(i, 0) = sum
sum = 0
Next
For j = 1 To y
For i = 1 To x
sum += roc(i, j) * (2 ^ (x - i))
Next
roc(0, j) = sum
sum = 0
Next
End Sub

```

โค้ดการสลับตำแหน่งผลรวมของแกน y (ในกรณีที่ y เป็นแกน x ทำเหมือนกันแต่สลับค่า x กับ y และทำสลับด้าน)

ตารางที่ ข – 5 โค้ดการสลับตำแหน่งผลรวมของแกน y

```

Private Sub cal_y()
Dim i As Integer, j As Integer, k As Integer
Dim x As Integer = Me.TextBox1.Text
Dim y As Integer = Me.TextBox2.Text
Dim temp As Double

```

ตารางที่ ข - 5 ได้การสลับตำแหน่งผลรวมของแกน y (ต่อ)

```

ReadMatrix()
For i = 2 To y
For j = y To i Step -1
If roc(0, j - 1) < roc(0, j) Then
For k = 0 To x
temp = roc(k, j - 1)
roc(k, j - 1) = roc(k, j)
roc(k, j) = temp
Next
temp = NOY(j - 1)
NOY(j - 1) = NOY(j)
NOY(j) = temp
End If
Next
Next
WriteMatrix()
End Sub

```

ได้แสดงค่าหลังสลับผลรวมแล้ว

ตารางที่ ข - 6 ได้แสดงค่าหลังสลับผลรวมแล้ว

```

Private Sub WriteMatrix()
Dim name(1) As String
Dim x As Integer, y As Integer
For Each c As Control In Me.GroupBox1.Controls
If TypeOf c Is TextBox Then
Dim t As TextBox = CType(c, TextBox)
name = Split(t.Tag.ToString, ",")

```

ตารางที่ ข - 6 ได้แสดงค่าหลังสลับผลรวมแล้ว (ต่อ)

```

x = name(0)
y = name(1)
t.Text = If(roc(x, y) = 0, "", roc(x, y))
End If
If TypeOf c Is Label Then
If c.Name.StartsWith("lbl") Then
Dim l As Label = CType(c, Label)
name = Split(l.Tag.ToString, ",")
x = name(0)
y = name(1)
If x = 0 Then
l.Text = NOY(y)
Else
l.Text = NOX(x)
End If
End If
End If
Next
End Sub

```

โค้ดของการทำซ้ำจนกว่าลำดับจะไม่เปลี่ยนแปลง

ตารางที่ ข - 7 โค้ดของการทำซ้ำจนกว่าลำดับจะไม่เปลี่ยนแปลง

```

While chkChanged
chkChanged = False

```

โค้ดของการคำนวณหาผลลัพธ์ที่ Button2 ในโปรแกรม ROC2 ประกอบไปด้วยฟังก์ชันย่อยดังต่อไปนี้

โค้ดการอ่านค่าในเมตริกซ์เพื่อนำมาเป็นไว้ใน Array 2 มิติ

ตารางที่ ข - 8 โค้ดการอ่านค่าในเมตริกซ์เพื่อนำมาเป็นไว้ใน Array 2 มิติในโปรแกรม ROC2

```

Private Sub ReadMatrixX()
Dim x As Integer = Me.TextBox1.Text
Dim y As Integer = Me.TextBox2.Text
Dim i As Integer, j As Integer, i1 As Integer, j1 As Integer
Dim x1 As Integer, y1 As Integer
Dim name(1) As String
Dim temp As Integer
Dim cell As Integer
Dim NoBefore As String = ""
Dim NoAfter As String = ""
ReDim NO(x)
ReDim roc(x, y)
For Each c As Control In Me.GroupBox1.Controls
If TypeOf c Is TextBox Then
Dim t As TextBox = CType(c, TextBox)
name = Split(t.Tag.ToString, ",")
x1 = name(0)
y1 = name(1)
roc(x1, y1) = If(t.Text = "", 0, t.Text)
End If
Next

```

ได้ผลการตรวจสอบว่าในเมตริกซ์ของไหนมีค่าเป็น 1

ตารางที่ ข - 9 ได้ผลการตรวจสอบว่าในเมตริกซ์ของไหนมีค่าเป็น 1

```

For i = y To 1 Step -1
For j = 1 To x
cell = NO(j).Cell
NO(j).Rank = roc(cell, i)
Next
Next

```

ได้ใช้สลับตำแหน่งเมื่อตรวจแล้วพบ 1 ในตำแหน่งนั้นในแกน x (ในกรณีที่เป็นแกน y ทำเหมือนกันแต่สลับค่า x กับ y และทำสลับด้าน)

ตารางที่ ข - 10 ได้ใช้สลับตำแหน่งเมื่อตรวจแล้วพบ 1 ในตำแหน่งนั้นในแกน x

```

For i1 = 2 To x
For j1 = x To i1 Step -1
If NO(j1 - 1).Rank < NO(j1).Rank Then
temp = NO(j1 - 1).No
NO(j1 - 1).No = NO(j1).No
NO(j1).No = temp
temp = NO(j1 - 1).Rank
NO(j1 - 1).Rank = NO(j1).Rank
NO(j1).Rank = temp
temp = NO(j1 - 1).Cell
NO(j1 - 1).Cell = NO(j1).Cell
NO(j1).Cell = temp
End If
Next
Next

```

ได้ทำการสร้างเมตริกซ์ขนาดเท่าตัวมันเองอีก 1 อัน เพื่อสลับตำแหน่ง โดยจะสร้างเป็น Temp ที่มี Array ขนาดเท่ากันแล้วทำการสลับที่ในเมตริกซ์ใหม่ แล้วนำเมตริกซ์ใหม่ที่ได้ไปแทนที่เมตริกซ์เก่า (ในกรณีที่ เป็นแกน y ทำเหมือนกันแต่สลับค่า x กับ y และทำสลับด้าน)

ตารางที่ ข - 11 ได้ทำการสร้างเมตริกซ์ขนาดเท่าตัวมันเองอีก 1 อัน

```
Private Sub SortMatrixX()
Dim i As Integer, j As Integer
Dim x As Integer = Me.TextBox1.Text
Dim y As Integer = Me.TextBox2.Text
Dim RocTemp(x, y) As Integer
Dim cell As Integer
For i = 0 To x
cell = NO(i).Cell
For j = 0 To y
RocTemp(i, j) = roc(cell, j)
Next
Next
roc = RocTemp
WriteMatrix()
End Sub
```

โค้ดเมื่อทำการแบ่งหน่วยการผลิตในโปรแกรม ROC1 และ ROC2

ตารางที่ ข - 12 ได้ดเมื่อทำการแบ่งหน่วยการผลิตในโปรแกรม ROC1 และ ROC2

```
Private Sub GroupBox1_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles GroupBox1.MouseDown
mx1 = e.X
my1 = e.Y
End Sub
```

ตารางที่ ข - 12 ได้ัดเมื่อทำการแบ่งหน่วยการผลิตในโปรแกรม ROC1 และ ROC2 (ต่อ)

```

Private Sub GroupBox1_MouseUp(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventArgs) Handles GroupBox1.MouseUp
mx2 = e.X
my2 = e.Y
If Not chkInput Then
Dim x, y As Integer
Dim temp As Integer
If mx1 > mx2 Then
temp = mx1
mx1 = mx2
mx2 = temp
End If
If my1 > my2 Then
temp = my1
my1 = my2
my2 = temp
End If
Dim RandomColor As Color = RandomQBColor()
For Each c As Control In Me.GroupBox1.Controls
If TypeOf c Is TextBox Then
If c.Name.StartsWith("txt") Then
Dim t As TextBox = CType(c, TextBox)
x = t.Location.X
y = t.Location.Y
If (x >= mx1 And x <= mx2) And (y >= my1 And y <= my2) Then
If t.BackColor.ToArgb.ToString = Color.White.ToArgb.ToString Then
t.BackColor = RandomColor
Else
t.BackColor = Color.White

```


ตารางที่ ข - 12 ได้ดเมื่อทำการแบ่งหน่วยการผลิตในโปรแกรม ROC1 และ ROC2 (ต่อ)

```
End If
End If
End If
End If
Next
End If
End Sub
```

โค้ดของการแบ่งจำนวนของ Exceptional Elements และจำนวนของ Voids เมื่อคลิกที่ Botton4 ในโปรแกรม ROC1 และ ROC2

ตารางที่ ข - 13 ได้ดของการแบ่งจำนวนของ Exceptional Elements และจำนวนของ Voids เมื่อคลิกที่ Botton4 ในโปรแกรม ROC1 และ ROC2

```
Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button4.Click
Dim name(1) As String
Dim ZeroIn As Integer = 0
Dim OneOut As Integer = 0
For Each c As Control In Me.GroupBox1.Controls
If TypeOf c Is TextBox Then
If c.Name.StartsWith("txt") Then
Dim t As TextBox = CType(c, TextBox)
If t.BackColor.ToArgb.ToString = Color.White.ToArgb.ToString Then
OneOut += If(t.Text = "1", 1, 0)
Else
ZeroIn += If(t.Text = "", 1, 0)
End If
End If
End If
```

ตารางที่ ข - 13 ได้ดของการแบ่งจำนวนของ Exceptional Elements และจำนวนของ Voids เมื่อคลิกที่ Botton4 ในโปรแกรม ROC1 และ ROC2 (ต่อ)

```
End If
Next
MsgBox("Number of Exceptional Elements = " & OneOut & " , Number of Voids = " &
ZeroIn)
End Sub
```





คู่มือการใช้งานโปรแกรมเพื่อแก้ปัญหาการสร้างหน่วยการผลิตในระบบการผลิตแบบ CMS โดยใช้วิธี ROC1 และ ROC2

โปรแกรมเพื่อแก้ปัญหาการสร้างหน่วยการผลิตในระบบการผลิตแบบ CMS โดยใช้วิธี ROC1 และ ROC2 เป็นโปรแกรมสำหรับช่วยแก้ปัญหาการสร้างหน่วยการผลิต โดยจะคำนึงถึงการผลิตระยะเวลาในการทำงาน และความถูกต้องเป็นหลัก พร้อมทั้งสามารถให้ผู้ใช้งานโปรแกรมสามารถกำหนดหน่วยการผลิตได้ด้วยตนเอง โดยแสดงค่าจำนวนของ Exceptional Elements และ จำนวนของ Voids เพื่อให้ผู้ใช้ได้พิจารณาผลลัพธ์

โดยในสื่อนี้จะอธิบายถึงขั้นตอนการทำงาน และส่วนประกอบต่าง ๆ ของโปรแกรมดังต่อไปนี้

1. ส่วนของ Main Menu เป็นส่วนที่ประกอบไปด้วย 2 ปุ่ม คือ ROC1 และ ROC2 โดยสามารถใช้เมาส์คลิกเลือกโปรแกรม ROC1 หรือ ROC2 ตามที่เราต้องการได้



รูปที่ ค - 1 แสดงหน้า Main Menu

2. ส่วนของโปรแกรม ROC1 และ ROC2 เมื่อเราคลิกเข้าไปใน ROC1 หรือ ROC2 แล้วเราจะพบกับ Menu ต่าง ๆ ดังนี้

Machines :	<input type="text"/>	จำนวนเครื่อง	จำนวนพาร์ท	และ Number of Exceptional Elements และ Number of Voids
Parts :	<input type="text"/>	Parts		

Machines

รูปที่ ค - 2 หน้าจอแสดงโปรแกรม ROC1 และ ROC2

3. ขั้นตอนการทำงานของโปรแกรม ROC1 และ ROC2

3.1 กรอกข้อมูลจำนวนเครื่องจักร และ ชิ้นส่วน โดยใช้เมาส์คลิกช่องว่างด้านหลัง

Machines : และ Parts : และกรอกข้อมูลลงไป

Machines :	6	สร้างเมตริกซ์	แสดงผลลัพธ์	แสดง Number of Exceptional Elements และ Number of Voids
Parts :	8			

Parts

Machines

รูปที่ ค - 3 ขั้นตอนการกรอกข้อมูลจำนวนเครื่องจักรและชิ้นส่วน

3.2 กดปุ่มสร้างเมตริกซ์ เพื่อสร้างเมตริกซ์จากข้อมูลจำนวนเครื่องจักรและชิ้นส่วน

Machines :	6	สร้างเมตริกซ์	แสดงผลลัพธ์	แสดง Number of Exceptional Elements และ Number of Voids
Parts :	8			

Parts

Machines

	1	2	3	4	5	6	7	8
1								
2								
3								
4								
5								
6								

รูปที่ ค - 4 หน้าจอเมื่อกดปุ่มสร้างเมตริกซ์

3.3 กรอกข้อมูลในเมตริกซ์โดยการเอาเมาส์ไปคลิกที่ช่องว่างในเมตริกซ์ก็จะปรากฏเป็นเลข 1 เมื่อคลิกอีกครั้งก็จะกลายเป็นช่องว่างเหมือนเดิม

Machines :	6	สร้างเมตริกซ์	แสดงผลลัพธ์	แสดง Number of Exceptional Elements และ Number of Voids
Parts :	8			

Parts

	1	2	3	4	5	6	7	8
Machines 1	1	1			1			1
2			1	1		1	1	
3	1	1	1		1	1		1
4				1			1	1
5			1	1			1	
6	1	1				1		

รูปที่ ค - 5 ขั้นตอนการกรอกข้อมูลในเมตริกซ์

3.4 กดปุ่มแสดงผลลัพธ์เพื่อให้โปรแกรมทำงานและหาผลลัพธ์

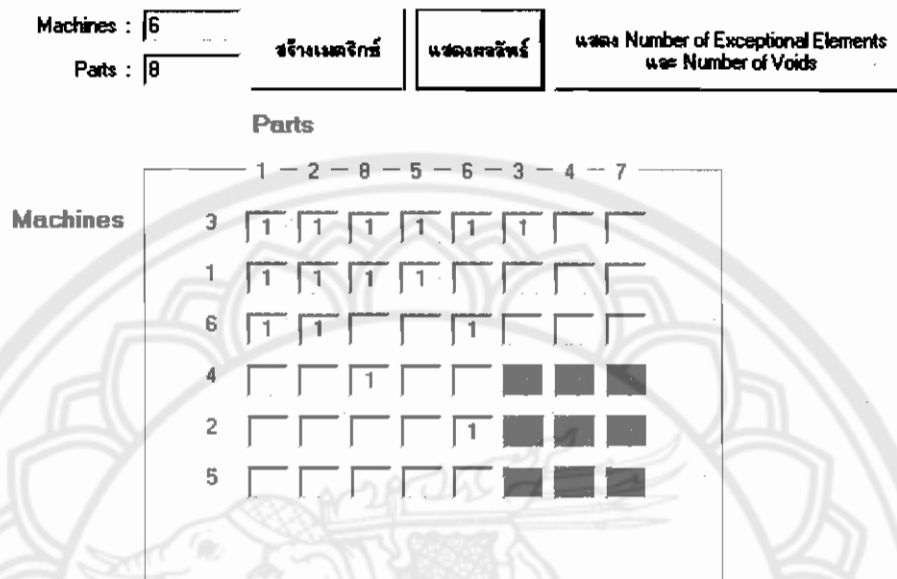
Machines :	6	สร้างเมตริกซ์	แสดงผลลัพธ์	แสดง Number of Exceptional Elements และ Number of Voids
Parts :	8			

Parts

	1	2	8	5	6	3	4	7
Machines 3	1	1	1	1	1	1		
1	1	1	1	1				
6	1	1			1			
4			1				1	1
2					1	1	1	1
5					1	1	1	

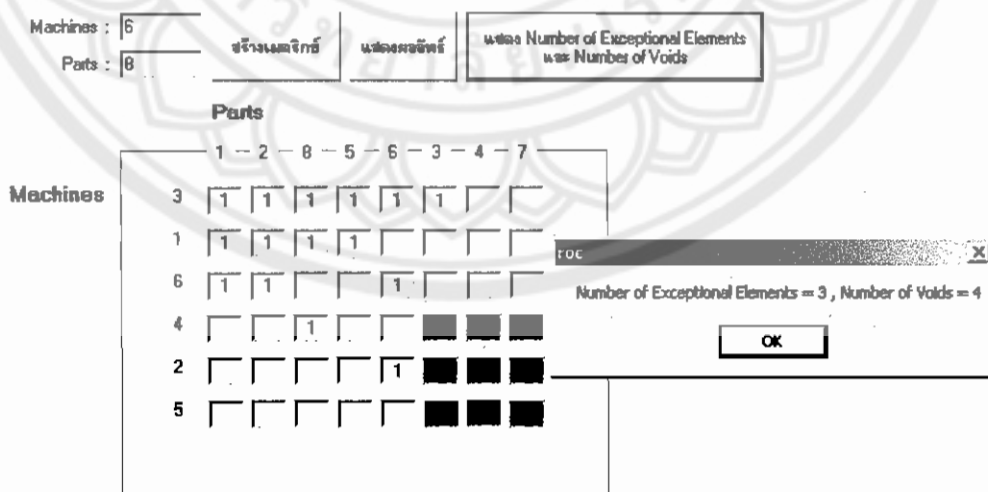
รูปที่ ค - 6 หน้าจอเมื่อกดปุ่มแสดงผลลัพธ์

3.5 ทำการแบ่งจำนวนหน่วยการผลิตโดยคลิกเมาส์ลากจากด้านบนซ้ายถึงด้านล่างขวา แล้วปล่อยตามตำแหน่งที่เราต้องการ และจะปรากฏเป็นสีตามหน่วยการผลิตที่เราได้คลิกเมาส์ลาก (ในลักษณะ Random สี)



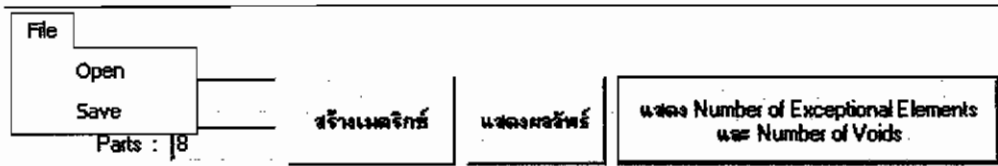
รูปที่ ค - 7 การแบ่งจำนวนหน่วยการผลิต

3.6 กดปุ่มแสดงจำนวนของ Exceptional Elements และจำนวนของ Voids เพื่อที่โปรแกรมจะแสดงผลลัพธ์



รูปที่ ค - 8 หน้าจอเมื่อกดปุ่ม แสดงจำนวนของ Exceptional Elements และจำนวนของ Voids

3.7 กดปุ่ม File เพื่อที่จะทำการเซฟ หรือ เปิดข้อมูลเก่า



Parts

	1	2	8	5	6	3	4	7
Machines								
3	1	1	1	1	1	1		
1	1	1	1	1				
6	1	1			1			
4			1					
2					1			
5								

รูปที่ ค - 9 หน้าจอเมื่อกดปุ่ม File