**โปรแกรมค้นหาตัวเลข**

**โดยใช้ ทฤษฎี Depth – First Search**

**ทฤษฎี Depth – First Search** เป็นวิธีการค้นหาแบบแนวลึกก่อน โดยจะค้นหาโหนดที่อยู่ในโครงสร้างต้นไม้ทีละระดับ เริ่มจากโหนดที่อยู่ระดับบนไปยังโหนดลูกที่อยู่ระดับล่าง เมื่อลงไปถึงโหนดสุดท้ายแล้ว ถ้ายังไม่พบโหนดเป้าหมายที่ต้องการให้ย้อนกลับขึ้นไปยังโหนดที่ผ่านมาล่าสุดหนึ่งระดับเพื่อมองหาโหนดที่เหลือ หากยังไม่พบโหนดเป้าหมายให้ทำเช่นนี้ไปเรื่อยๆ จนกว่าจะพบโหนดเป้าหมายที่ต้องการการค้นหาจะใช้โครงสร้างข้อมูลแบบสแต็ก (Stack) เพราะทำงานแบบ LIFO

**โครงสร้างข้อมูลแบบ Stack** มีโครงสร้างข้อมูลแบบ Last-in-First-Out คือ ข้อมูลที่เข้ามาเก็บใน Stack ทีหลังจะเป็นข้อมูลที่ออกจาก Stack ก่อน เช่น การเรียงซ้อนกันของจาน ในการหยิบจานครั้งแรกจะต้องหยิบจานที่อยู่บนสุดก่อน ซึ่งเป็นจานใบล่าสุดที่นำมาวาง เป็นต้น

**โครงสร้างข้อมูล Stack**

Stack เป็นโครงสร้างข้อมูล ที่มีคุณสมบัติที่ว่าการเพิ่มหรือลบข้อมูลในสแต็กจะกระทำได้ทางเดียว และข้อมูลที่เข้ามาที่หลังจะต้องออกไปก่อน

**1. ลักษณะสำคัญของโครงสร้างข้อมูล Stack**

- โครงสร้างข้อมูลเป็นแบบเชิงเส้น ลักษณะโครงสร้างจัดเรียงต่อเนื่องกันไป

- มีลักษณะโครงสร้างที่ไม่ตายตัว สามารถปรับเปลี่ยนจำนวนสมาชิกได้

- นำข้อมูลเข้าและดึงข้อมูลออกได้ สามารถจะทำการ Push และ Pop

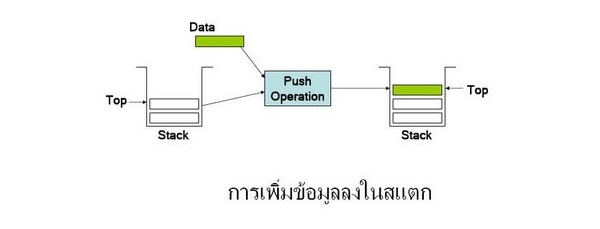
- นำข้อมูลเข้าและดึงข้อมูลออกเป็นลำดับ ไม่ข้ามหรือกระโดดไปเอาข้อมูลใดข้อมูลหนึ่งก่อน

- มีการจัดการนำเข้าและดึงข้อมูลในตำแหน่งบนสุด การนำเข้าข้อมูลและการดึงข้อมูลออกต้องทำในตำแหน่งบนสุดเท่านั้น

**2. พื้นฐานการดำเนินการของโครงสร้างข้อมูล Stack**

**Push หรือการนำข้อมูลเข้า**

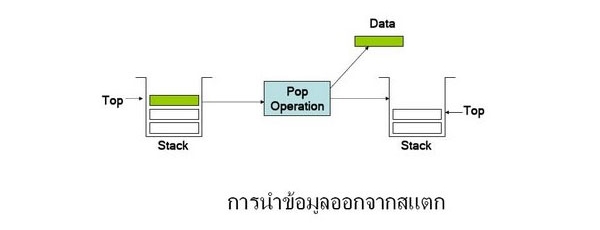
Push คือ การนำข้อมูลใส่ลงไปในสแต็ก เช่น สแต็ก s ต้องการใส่ข้อมูล i ในสแต็กจะได้ Push (s,i) คือ ใส่ข้อมูล i ลงไปที่ท็อปของสแต็ก s ในการเพิ่มข้อมูลลงในสแต็ก จะต้องทำการตรวจสอบว่าสแต็กเต็มหรือไม่ ถ้าไม่เต็มก็ สามารถเพิ่มข้อมูลลงไปในสแต็กได้แล้วปรับตัวชี้ตำแหน่งให้ไปชี้ที่ตำแหน่งข้อมูลใหม่ ถ้าสแต็กเต็ม (Stack Overflow) ก็จะไม่สามารถเพิ่มข้อมูลเข้าไปในสแต็กได้อีก



**Pop หรือการนำออกข้อมูล**

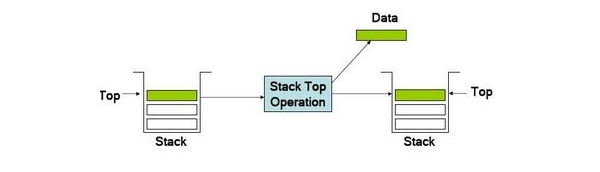
Pop คือการนำข้อมูลออกจากส่วนบนสุดของสแต็ก เช่น ต้องการนำข้อมูลออกจาก สแต็ก s ไปไว้ที่ตัวแปร i จะได้ i = Pop (s) การนำข้อมูลออกจากสแต็ก ถ้าสแต็กมีสมาชิกเพียง 1 ตัวแล้วนำสมาชิกออกจาก

สแต็ก จะเกิดสภาวะสแต็กว่าง (Stack Empty) คือไม่มีสมาชิกอยู่ในสแต็กเลยแต่ถ้าไม่มีสมาชิกในสแต็กแล้วทำการ Pop สแต็กจะทำให้เกิดความผิดพลาดที่เรียกว่า Stack Underflow เพราะฉะนั้นก่อนนำข้อมูลออกจากสแต็กจะต้องตรวจสอบก่อนว่าสแต็กว่างหรือเปล่าจึงจะนำข้อมูลออกจากสแต็กได้ และ ปรับตัวชี้ตำแหน่งให้ไปชี้ตำแหน่งของข้อมูลที่ต่อจากข้อมูลที่ถูกนำออกไป



**Top หรือตำแหน่งบนสุด**

Stack Top เป็นการคัดลอกข้อมูลที่อยู่บนสุดของสแต็กแต่ไม่ได้นำเอาข้อมูลนั้นออกจากสแต็ก ซึ่ง Top จะเกิดข้อผิดพลาดเช่นเดียวกับการ Pop คือ Stack Underflow เมื่อสแต็กนั้นเกิดการว่าง



**Stack มีพร็อพเพอร์ตี้และเมธอดพื้นฐานที่ควรรู้ดังนี้**

Push เพิ่มข้อมูลเข้าไปที่ตำแหน่งบนสุดของ Stack

Pop เอาข้อมูลออกจากตำแหน่งบนสุดของ Stack

Peek รีเทิร์นค่าข้อมูลบนสุดของ Stack

Count จำนวนสมาชิกของ Stack

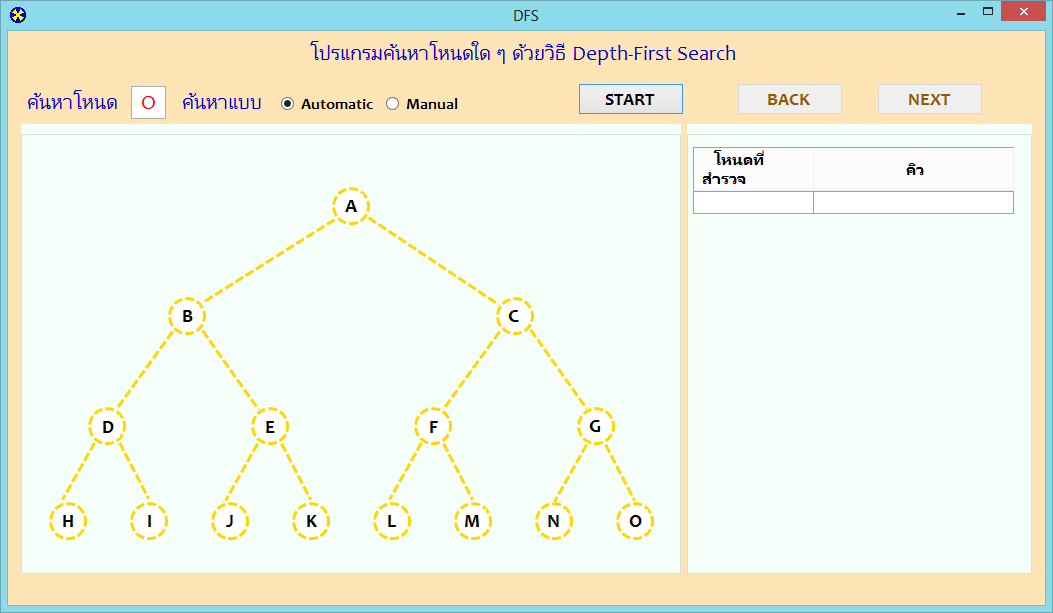
Contains ตรวจสอบว่ามีข้อมูลที่ต้องการใน Stack หรือไม่

To Array เป็นเมธอดที่ใช้ก็อปปี้ทุกค่าใน Stack ไปสู่อาร์เรย์

Clear เป็นเมธอดที่ใช้ล้างข้อมูลทุกตัวออกจาก Stack

แสดงแผนการทำงานของโปรแกรม

**1.ออกแบบโปรแกรมดังนี้**



**ตัวอย่างโค๊ด**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace Project\_Depth\_First\_Search

{

public partial class Form1 : Form

{

int round = 0;

string s = "";

public Form1()

{

InitializeComponent();

pa1.BorderColor = Color.MintCream; pa2.BorderColor = Color.MintCream; pa3.BorderColor = Color.MintCream;

pb1.BorderColor = Color.MintCream; pb2.BorderColor = Color.MintCream; pb3.BorderColor = Color.MintCream;

pc1.BorderColor = Color.MintCream; pc2.BorderColor = Color.MintCream; pc3.BorderColor = Color.MintCream;

pd1.BorderColor = Color.MintCream; pd2.BorderColor = Color.MintCream; pd3.BorderColor = Color.MintCream;

pe1.BorderColor = Color.MintCream; pe2.BorderColor = Color.MintCream; pe3.BorderColor = Color.MintCream;

pf1.BorderColor = Color.MintCream; pf2.BorderColor = Color.MintCream; pf3.BorderColor = Color.MintCream;

pg1.BorderColor = Color.MintCream; pg2.BorderColor = Color.MintCream; pg3.BorderColor = Color.MintCream;

ph1.BorderColor = Color.MintCream; ph2.BorderColor = Color.MintCream; ph3.BorderColor = Color.MintCream;

pi1.BorderColor = Color.MintCream; pi2.BorderColor = Color.MintCream; pi3.BorderColor = Color.MintCream;

pj1.BorderColor = Color.MintCream; pj2.BorderColor = Color.MintCream; pj3.BorderColor = Color.MintCream;

pk1.BorderColor = Color.MintCream; pk2.BorderColor = Color.MintCream; pk3.BorderColor = Color.MintCream;

pl1.BorderColor = Color.MintCream; pl2.BorderColor = Color.MintCream; pl3.BorderColor = Color.MintCream;

pm1.BorderColor = Color.MintCream; pm2.BorderColor = Color.MintCream; pm3.BorderColor = Color.MintCream;

pn1.BorderColor = Color.MintCream; pn2.BorderColor = Color.MintCream; pn3.BorderColor = Color.MintCream;

po1.BorderColor = Color.MintCream; po2.BorderColor = Color.MintCream; po3.BorderColor = Color.MintCream;

start.Enabled = true ;back.Enabled = false; next.Enabled = false;

}

private void Form1\_Load(object sender, EventArgs e)

{

}

private void start\_Click(object sender, EventArgs e)

{

String c = TargetBox.Text;

switch (c)

{

case "a": TargetBox.Text = "A"; break;

case "b": TargetBox.Text = "B"; break;

case "c": TargetBox.Text = "C"; break;

case "d": TargetBox.Text = "D"; break;

case "e": TargetBox.Text = "E"; break;

case "f": TargetBox.Text = "F"; break;

case "g": TargetBox.Text = "G"; break;

case "h": TargetBox.Text = "H"; break;

case "i": TargetBox.Text = "I"; break;

case "j": TargetBox.Text = "J"; break;

case "k": TargetBox.Text = "K"; break;

case "l": TargetBox.Text = "L"; break;

case "m": TargetBox.Text = "M"; break;

case "n": TargetBox.Text = "N"; break;

case "o": TargetBox.Text = "O"; break;

}

if (automatic.Checked == true && start.Text.Equals("START"))

{

start.Enabled = false;

runTimer.Enabled = true;

}

else if (automatic.Checked == true && start.Text.Equals("FINISH"))

{

TargetBox.Enabled = true;

start.Text = "START";

Del\_Search();

}

else {

if (start.Text.Equals("START"))

{

TargetBox.Enabled = false;

start.Text = "RESET";

next.Enabled = true;

Next\_Search();

}

else if (start.Text.Equals("RESET"))

{

TargetBox.Enabled = true;

start.Text = "START";

//back.Enabled = false;

//next.Enabled = false;

Del\_Search();

}

else

{

TargetBox.Enabled = true;

start.Text = "START";

Del\_Search();

}

}

}

private void back\_Click(object sender, EventArgs e)

{

Back\_Search();

}

private void next\_Click(object sender, EventArgs e)

{

String s = tlist[0, round].Value + "";

//lblStart.Text = s;

back.Enabled = true;

Next\_Search();

}

private void Del\_Search()

{

int chk = tlist.Rows.Count;

//MessageBox.Show("Row of tlist " + chk + " row(s)", "Finish", MessageBoxButtons.OK, MessageBoxIcon.Information);

for (int i = 1; i < chk;i++) {

tlist.Rows.RemoveAt(0);

}

node\_a.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_a.BorderColor = Color.Gold;node\_a.FillColor = Color.White;

node\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_b.BorderColor = Color.Gold;node\_b.FillColor = Color.White;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_c.BorderColor = Color.Gold;node\_c.FillColor = Color.White;

node\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_d.BorderColor = Color.Gold;node\_d.FillColor = Color.White;

node\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_e.BorderColor = Color.Gold;node\_e.FillColor = Color.White;

node\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_f.BorderColor = Color.Gold;node\_f.FillColor = Color.White;

node\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_g.BorderColor = Color.Gold;node\_g.FillColor = Color.White;

node\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_h.BorderColor = Color.Gold;node\_h.FillColor = Color.White;

node\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_i.BorderColor = Color.Gold;node\_i.FillColor = Color.White;

node\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_j.BorderColor = Color.Gold;node\_j.FillColor = Color.White;

node\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_k.BorderColor = Color.Gold;node\_k.FillColor = Color.White;

node\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_l.BorderColor = Color.Gold;node\_l.FillColor = Color.White;

node\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_m.BorderColor = Color.Gold;node\_m.FillColor = Color.White;

node\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_n.BorderColor = Color.Gold;node\_n.FillColor = Color.White;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;node\_o.BorderColor = Color.Gold;node\_o.FillColor = Color.White;

pa1.BorderColor = Color.MintCream; pa2.BorderColor = Color.MintCream; pa3.BorderColor = Color.MintCream;

pb1.BorderColor = Color.MintCream; pb2.BorderColor = Color.MintCream; pb3.BorderColor = Color.MintCream;

pc1.BorderColor = Color.MintCream; pc2.BorderColor = Color.MintCream; pc3.BorderColor = Color.MintCream;

pd1.BorderColor = Color.MintCream; pd2.BorderColor = Color.MintCream; pd3.BorderColor = Color.MintCream;

pe1.BorderColor = Color.MintCream; pe2.BorderColor = Color.MintCream; pe3.BorderColor = Color.MintCream;

pf1.BorderColor = Color.MintCream; pf2.BorderColor = Color.MintCream; pf3.BorderColor = Color.MintCream;

pg1.BorderColor = Color.MintCream; pg2.BorderColor = Color.MintCream; pg3.BorderColor = Color.MintCream;

ph1.BorderColor = Color.MintCream; ph2.BorderColor = Color.MintCream; ph3.BorderColor = Color.MintCream;

pi1.BorderColor = Color.MintCream; pi2.BorderColor = Color.MintCream; pi3.BorderColor = Color.MintCream;

pj1.BorderColor = Color.MintCream; pj2.BorderColor = Color.MintCream; pj3.BorderColor = Color.MintCream;

pk1.BorderColor = Color.MintCream; pk2.BorderColor = Color.MintCream; pk3.BorderColor = Color.MintCream;

pl1.BorderColor = Color.MintCream; pl2.BorderColor = Color.MintCream; pl3.BorderColor = Color.MintCream;

pm1.BorderColor = Color.MintCream; pm2.BorderColor = Color.MintCream; pm3.BorderColor = Color.MintCream;

pn1.BorderColor = Color.MintCream; pn2.BorderColor = Color.MintCream; pn3.BorderColor = Color.MintCream;

po1.BorderColor = Color.MintCream; po2.BorderColor = Color.MintCream; po3.BorderColor = Color.MintCream;

line\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_b.BorderColor = Color.Gold;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_c.BorderColor = Color.Gold;

line\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_d.BorderColor = Color.Gold;

line\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_e.BorderColor = Color.Gold;

line\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_f.BorderColor = Color.Gold;

line\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_g.BorderColor = Color.Gold;

line\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_h.BorderColor = Color.Gold;

line\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_i.BorderColor = Color.Gold;

line\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_j.BorderColor = Color.Gold;

line\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_k.BorderColor = Color.Gold;

line\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_l.BorderColor = Color.Gold;

line\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_m.BorderColor = Color.Gold;

line\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_n.BorderColor = Color.Gold;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;line\_o.BorderColor = Color.Gold;

lb\_a.BackColor = Color.White;lb\_b.BackColor = Color.White;lb\_c.BackColor = Color.White;

lb\_d.BackColor = Color.White;lb\_e.BackColor = Color.White;lb\_f.BackColor = Color.White;

lb\_g.BackColor = Color.White;lb\_h.BackColor = Color.White;lb\_i.BackColor = Color.White;

lb\_j.BackColor = Color.White;lb\_k.BackColor = Color.White;lb\_l.BackColor = Color.White;

lb\_m.BackColor = Color.White;lb\_n.BackColor = Color.White;lb\_o.BackColor = Color.White;

round = 0;

}

private void Back\_Search()

{

String s = tlist[0, round - 1].Value + "";

//lblStart.Text = s;

if (s == "B") //ลูกศรชี้ที่ A

{

tlist.Rows.RemoveAt(round-1);

pb1.BorderColor = Color.MintCream; pb2.BorderColor = Color.MintCream; pb3.BorderColor = Color.MintCream;

pa1.BorderColor = Color.Red; pa2.BorderColor = Color.Red; pa3.BorderColor = Color.Red;

node\_a.FillColor = Color.White;

lb\_a.BackColor = Color.White;

node\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_b.BorderColor = Color.Black;

line\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_b.BorderColor = Color.Black;

node\_b.FillColor = Color.White;

lb\_b.BackColor = Color.White;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_c.BorderColor = Color.Black;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_c.BorderColor = Color.Black;

node\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_d.BorderColor = Color.Gold;

line\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_d.BorderColor = Color.Gold;

node\_d.FillColor = Color.White;

lb\_d.BackColor = Color.White;

node\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_e.BorderColor = Color.Gold;

line\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_e.BorderColor = Color.Gold;

node\_e.FillColor = Color.White;

lb\_e.BackColor = Color.White;

back.Enabled = false;

}

else if (s == "C") //ลูกศรชี้ที่ K

{

tlist.Rows.RemoveAt(round - 1);

pc1.BorderColor = Color.MintCream; pc2.BorderColor = Color.MintCream; pc3.BorderColor = Color.MintCream;

pk1.BorderColor = Color.Red; pk2.BorderColor = Color.Red; pk3.BorderColor = Color.Red;

node\_k.FillColor = Color.White;

lb\_k.BackColor = Color.White;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_c.BorderColor = Color.Black;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_c.BorderColor = Color.Black;

node\_c.FillColor = Color.White;

lb\_c.BackColor = Color.White;

node\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_f.BorderColor = Color.Gold;

line\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_f.BorderColor = Color.Gold;

node\_f.FillColor = Color.White;

lb\_f.BackColor = Color.White;

node\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_g.BorderColor = Color.Gold;

line\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_g.BorderColor = Color.Gold;

node\_g.FillColor = Color.White;

lb\_g.BackColor = Color.White;

}

else if (s == "D") //ลูกศรชี้ที่ B

{

tlist.Rows.RemoveAt(round - 1);

pd1.BorderColor = Color.MintCream; pd2.BorderColor = Color.MintCream; pd3.BorderColor = Color.MintCream;

pb1.BorderColor = Color.Red; pb2.BorderColor = Color.Red; pb3.BorderColor = Color.Red;

node\_b.FillColor = Color.White;

lb\_b.BackColor = Color.White;

node\_d.FillColor = Color.White;

lb\_d.BackColor = Color.White;

node\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_d.BorderColor = Color.Black;

line\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_d.BorderColor = Color.Black;

node\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_h.BorderColor = Color.Gold;

line\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_h.BorderColor = Color.Gold;

node\_h.FillColor = Color.White;

lb\_h.BackColor = Color.White;

node\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_i.BorderColor = Color.Gold;

line\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_i.BorderColor = Color.Gold;

node\_i.FillColor = Color.White;

lb\_i.BackColor = Color.White;

}

else if (s == "E") //ลูกศรชี้ที่ i

{

tlist.Rows.RemoveAt(round - 1);

pe1.BorderColor = Color.MintCream; pe2.BorderColor = Color.MintCream; pe3.BorderColor = Color.MintCream;

pi1.BorderColor = Color.Red; pi2.BorderColor = Color.Red; pi3.BorderColor = Color.Red;

node\_i.FillColor = Color.White;

lb\_i.BackColor = Color.White;

node\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_e.BorderColor = Color.Black;

line\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_e.BorderColor = Color.Black;

node\_e.FillColor = Color.White;

lb\_e.BackColor = Color.White;

node\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_j.BorderColor = Color.Gold;

line\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_j.BorderColor = Color.Gold;

node\_j.FillColor = Color.White;

lb\_j.BackColor = Color.White;

node\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_k.BorderColor = Color.Gold;

line\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_k.BorderColor = Color.Gold;

node\_k.FillColor = Color.White;

lb\_k.BackColor = Color.White;

}

else if (s == "F") //ลูกศรชี้ที่ C

{

tlist.Rows.RemoveAt(round - 1);

pf1.BorderColor = Color.MintCream; pf2.BorderColor = Color.MintCream; pf3.BorderColor = Color.MintCream;

pc1.BorderColor = Color.Red; pc2.BorderColor = Color.Red; pc3.BorderColor = Color.Red;

node\_c.FillColor = Color.White;

lb\_c.BackColor = Color.White;

node\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_f.BorderColor = Color.Black;

line\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_f.BorderColor = Color.Black;

node\_f.FillColor = Color.White;

lb\_f.BackColor = Color.White;

node\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_l.BorderColor = Color.Gold;

line\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_l.BorderColor = Color.Gold;

node\_l.FillColor = Color.White;

lb\_l.BackColor = Color.White;

node\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_m.BorderColor = Color.Gold;

line\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_m.BorderColor = Color.Gold;

node\_m.FillColor = Color.White;

lb\_m.BackColor = Color.White;

}

else if (s == "G") //ลูกศรชี้ที่ M

{

tlist.Rows.RemoveAt(round - 1);

pg1.BorderColor = Color.MintCream; pg2.BorderColor = Color.MintCream; pg3.BorderColor = Color.MintCream;

pm1.BorderColor = Color.Red; pm2.BorderColor = Color.Red; pm3.BorderColor = Color.Red;

node\_m.FillColor = Color.White;

lb\_m.BackColor = Color.White;

node\_g.FillColor = Color.White;

lb\_g.BackColor = Color.White;

node\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_g.BorderColor = Color.Black;

line\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_g.BorderColor = Color.Black;

node\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_n.BorderColor = Color.Gold;

line\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_n.BorderColor = Color.Gold;

node\_n.FillColor = Color.White;

lb\_n.BackColor = Color.White;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_o.BorderColor = Color.Gold;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_o.BorderColor = Color.Gold;

node\_o.FillColor = Color.White;

lb\_o.BackColor = Color.White;

}

else if (s == "H") //ลูกศรชี้ที่ D

{

tlist.Rows.RemoveAt(round - 1);

ph1.BorderColor = Color.MintCream; ph2.BorderColor = Color.MintCream; ph3.BorderColor = Color.MintCream;

pd1.BorderColor = Color.Red; pd2.BorderColor = Color.Red; pd3.BorderColor = Color.Red;

node\_d.FillColor = Color.White;

lb\_d.BackColor = Color.White;

node\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_h.BorderColor = Color.Black;

line\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_h.BorderColor = Color.Black;

node\_h.FillColor = Color.White;

lb\_h.BackColor = Color.White;

}

else if (s == "I") //ลูกศรชี้ที่ H

{

tlist.Rows.RemoveAt(round - 1);

pi1.BorderColor = Color.MintCream; pi2.BorderColor = Color.MintCream; pi3.BorderColor = Color.MintCream;

ph1.BorderColor = Color.Red; ph2.BorderColor = Color.Red; ph3.BorderColor = Color.Red;

node\_h.FillColor = Color.White;

lb\_h.BackColor = Color.White;

line\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_i.BorderColor = Color.Black;

node\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_i.BorderColor = Color.Black;

node\_i.FillColor = Color.White;

lb\_i.BackColor = Color.White;

}

else if (s == "J") //ลูกศรชี้ที่ E

{

tlist.Rows.RemoveAt(round - 1);

pj1.BorderColor = Color.MintCream; pj2.BorderColor = Color.MintCream; pj3.BorderColor = Color.MintCream;

pe1.BorderColor = Color.Red; pe2.BorderColor = Color.Red; pe3.BorderColor = Color.Red;

node\_e.FillColor = Color.White;

lb\_e.BackColor = Color.White;

line\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_j.BorderColor = Color.Black;

node\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_j.BorderColor = Color.Black;

node\_j.FillColor = Color.White;

lb\_j.BackColor = Color.White;

}

else if (s == "K") //ลูกศรชี้ที่ J

{

tlist.Rows.RemoveAt(round - 1);

pk1.BorderColor = Color.MintCream; pk2.BorderColor = Color.MintCream; pk3.BorderColor = Color.MintCream;

pj1.BorderColor = Color.Red; pj2.BorderColor = Color.Red; pj3.BorderColor = Color.Red;

node\_j.FillColor = Color.White;

lb\_j.BackColor = Color.White;

line\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_k.BorderColor = Color.Black;

node\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_k.BorderColor = Color.Black;

node\_k.FillColor = Color.White;

lb\_k.BackColor = Color.White;

}

else if (s == "L") //ลูกศรชี้ที่ F

{

tlist.Rows.RemoveAt(round - 1);

pl1.BorderColor = Color.MintCream; pl2.BorderColor = Color.MintCream; pl3.BorderColor = Color.MintCream;

pf1.BorderColor = Color.Red; pf2.BorderColor = Color.Red; pf3.BorderColor = Color.Red;

node\_f.FillColor = Color.White;

lb\_f.BackColor = Color.White;

line\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_l.BorderColor = Color.Black;

node\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_l.BorderColor = Color.Black;

node\_l.FillColor = Color.White;

lb\_l.BackColor = Color.White;

}

else if (s == "M") //ลูกศรชี้ที่ L

{

tlist.Rows.RemoveAt(round - 1);

pm1.BorderColor = Color.MintCream; pm2.BorderColor = Color.MintCream; pm3.BorderColor = Color.MintCream;

pl1.BorderColor = Color.Red; pl2.BorderColor = Color.Red; pl3.BorderColor = Color.Red;

node\_l.FillColor = Color.White;

lb\_l.BackColor = Color.White;

node\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_m.BorderColor = Color.Black;

line\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_m.BorderColor = Color.Black;

node\_m.FillColor = Color.White;

lb\_m.BackColor = Color.White;

}

else if (s == "N") //ลูกศรชี้ที่ G

{

tlist.Rows.RemoveAt(round - 1);

pn1.BorderColor = Color.MintCream; pn2.BorderColor = Color.MintCream; pn3.BorderColor = Color.MintCream;

pg1.BorderColor = Color.Red; pg2.BorderColor = Color.Red; pg3.BorderColor = Color.Red;

node\_g.FillColor = Color.White;

lb\_g.BackColor = Color.White;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_o.BorderColor = Color.Black;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_o.BorderColor = Color.Black;

node\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_n.BorderColor = Color.Black;

line\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_n.BorderColor = Color.Black;

}

else if (s == "O") //ลูกศรชี้ที่ N

{

tlist.Rows.RemoveAt(round - 1);

po1.BorderColor = Color.MintCream; po2.BorderColor = Color.MintCream; po3.BorderColor = Color.MintCream;

pn1.BorderColor = Color.Red; pn2.BorderColor = Color.Red; pn3.BorderColor = Color.Red;

node\_n.FillColor = Color.White;

lb\_n.BackColor = Color.White;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_o.BorderColor = Color.Black;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_o.BorderColor = Color.Black;

}

round--;

}

private void Next\_Search(){

try

{

if (tlist[0, 0].Value + "" == "")

{

pa1.BorderColor = Color.Red; pa2.BorderColor = Color.Red; pa3.BorderColor = Color.Red;

node\_a.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_a.BorderColor = Color.Black;

tlist.Rows.Add("A", "C B");

node\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_b.BorderColor = Color.Black;

line\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_b.BorderColor = Color.Black;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_c.BorderColor = Color.Black;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_c.BorderColor = Color.Black;

}

else

{

s = tlist[1, round - 1].Value + "";

if (s.Substring(s.Length - 1, 1) == "B")

{

tlist.Rows.Add("B", "C E D");

pa1.BorderColor = Color.MintCream; pa2.BorderColor = Color.MintCream; pa3.BorderColor = Color.MintCream; //เปลี่ยนสีลูกศรโหนด a ให้เข้ากับสีพื้นหลัง

pb1.BorderColor = Color.Red; pb2.BorderColor = Color.Red; pb3.BorderColor = Color.Red; //เปลี่ยนสีลูกศรโหนด b ให้มองเห็น

node\_a.FillColor = Color.DarkGray;

lb\_a.BackColor = Color.DarkGray;

line\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

line\_b.BorderColor = Color.Black;

node\_b.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_b.BorderColor = Color.Black;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_c.BorderColor = Color.Black;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_c.BorderColor = Color.Black;

node\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_d.BorderColor = Color.Black;

line\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_d.BorderColor = Color.Black;

node\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_e.BorderColor = Color.Black;

line\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_e.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "D")

{

tlist.Rows.Add("D", "C E I H");

pb1.BorderColor = Color.MintCream; pb2.BorderColor = Color.MintCream; pb3.BorderColor = Color.MintCream;

pd1.BorderColor = Color.Red; pd2.BorderColor = Color.Red; pd3.BorderColor = Color.Red;

node\_b.FillColor = Color.DarkGray;

lb\_b.BackColor = Color.DarkGray;

line\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_d.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_d.BorderColor = Color.Black;

node\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_h.BorderColor = Color.Black;

line\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_h.BorderColor = Color.Black;

node\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_i.BorderColor = Color.Black;

line\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_i.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "H")

{

tlist.Rows.Add("H", "C E I");

pd1.BorderColor = Color.MintCream; pd2.BorderColor = Color.MintCream; pd3.BorderColor = Color.MintCream;

ph1.BorderColor = Color.Red; ph2.BorderColor = Color.Red; ph3.BorderColor = Color.Red;

node\_d.FillColor = Color.DarkGray;

lb\_d.BackColor = Color.DarkGray;

line\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_h.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_h.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "I")

{

tlist.Rows.Add("I", "C E");

ph1.BorderColor = Color.MintCream; ph2.BorderColor = Color.MintCream; ph3.BorderColor = Color.MintCream;

pi1.BorderColor = Color.Red; pi2.BorderColor = Color.Red; pi3.BorderColor = Color.Red;

node\_h.FillColor = Color.DarkGray;

lb\_h.BackColor = Color.DarkGray;

line\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_i.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_i.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "E")

{

tlist.Rows.Add("E", "C K J");

pi1.BorderColor = Color.MintCream; pi2.BorderColor = Color.MintCream; pi3.BorderColor = Color.MintCream;

pe1.BorderColor = Color.Red; pe2.BorderColor = Color.Red; pe3.BorderColor = Color.Red;

node\_i.FillColor = Color.DarkGray;

lb\_i.BackColor = Color.DarkGray;

line\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_e.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_e.BorderColor = Color.Black;

node\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_j.BorderColor = Color.Black;

line\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_j.BorderColor = Color.Black;

node\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_k.BorderColor = Color.Black;

line\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_k.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "J")

{

tlist.Rows.Add("J", "C K");

pe1.BorderColor = Color.MintCream; pe2.BorderColor = Color.MintCream; pe3.BorderColor = Color.MintCream;

pj1.BorderColor = Color.Red; pj2.BorderColor = Color.Red; pj3.BorderColor = Color.Red;

node\_e.FillColor = Color.DarkGray;

lb\_e.BackColor = Color.DarkGray;

line\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_j.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_j.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "K")

{

tlist.Rows.Add("K", "C");

pj1.BorderColor = Color.MintCream; pj2.BorderColor = Color.MintCream; pj3.BorderColor = Color.MintCream;

pk1.BorderColor = Color.Red; pk2.BorderColor = Color.Red; pk3.BorderColor = Color.Red;

node\_j.FillColor = Color.DarkGray;

lb\_j.BackColor = Color.DarkGray;

line\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_k.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_k.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "C")

{

tlist.Rows.Add("C", "G F");

pk1.BorderColor = Color.MintCream; pk2.BorderColor = Color.MintCream; pk3.BorderColor = Color.MintCream;

pc1.BorderColor = Color.Red; pc2.BorderColor = Color.Red; pc3.BorderColor = Color.Red;

node\_k.FillColor = Color.DarkGray;

lb\_k.BackColor = Color.DarkGray;

line\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_c.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_c.BorderColor = Color.Black;

node\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_f.BorderColor = Color.Black;

line\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_f.BorderColor = Color.Black;

node\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_g.BorderColor = Color.Black;

line\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_g.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "F")

{

tlist.Rows.Add("F", "G M L");

pc1.BorderColor = Color.MintCream; pc2.BorderColor = Color.MintCream; pc3.BorderColor = Color.MintCream;

pf1.BorderColor = Color.Red; pf2.BorderColor = Color.Red; pf3.BorderColor = Color.Red;

node\_c.FillColor = Color.DarkGray;

lb\_c.BackColor = Color.DarkGray;

line\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_f.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_f.BorderColor = Color.Black;

node\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_l.BorderColor = Color.Black;

line\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_l.BorderColor = Color.Black;

node\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_m.BorderColor = Color.Black;

line\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_m.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "L")

{

tlist.Rows.Add("L", "G M");

pf1.BorderColor = Color.MintCream; pf2.BorderColor = Color.MintCream; pf3.BorderColor = Color.MintCream;

pl1.BorderColor = Color.Red; pl2.BorderColor = Color.Red; pl3.BorderColor = Color.Red;

node\_f.FillColor = Color.DarkGray;

lb\_f.BackColor = Color.DarkGray;

line\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_l.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_l.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "M")

{

tlist.Rows.Add("M", "G");

pl1.BorderColor = Color.MintCream; pl2.BorderColor = Color.MintCream; pl3.BorderColor = Color.MintCream;

pm1.BorderColor = Color.Red; pm2.BorderColor = Color.Red; pm3.BorderColor = Color.Red;

node\_l.FillColor = Color.DarkGray;

lb\_l.BackColor = Color.DarkGray;

line\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_m.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_m.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "G")

{

tlist.Rows.Add("G", "O N");

pm1.BorderColor = Color.MintCream; pm2.BorderColor = Color.MintCream; pm3.BorderColor = Color.MintCream;

pg1.BorderColor = Color.Red; pg2.BorderColor = Color.Red; pg3.BorderColor = Color.Red;

node\_m.FillColor = Color.DarkGray;

lb\_m.BackColor = Color.DarkGray;

line\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_g.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_g.BorderColor = Color.Black;

node\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_n.BorderColor = Color.Black;

line\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_n.BorderColor = Color.Black;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

node\_o.BorderColor = Color.Black;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Dash;

line\_o.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "N")

{

tlist.Rows.Add("N", "O");

pg1.BorderColor = Color.MintCream; pg2.BorderColor = Color.MintCream; pg3.BorderColor = Color.MintCream;

pn1.BorderColor = Color.Red; pn2.BorderColor = Color.Red; pn3.BorderColor = Color.Red;

node\_g.FillColor = Color.DarkGray;

lb\_g.BackColor = Color.DarkGray;

line\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_n.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_n.BorderColor = Color.Black;

}

else if (s.Substring(s.Length - 1, 1) == "O")

{

tlist.Rows.Add("O", "");

pn1.BorderColor = Color.MintCream; pn2.BorderColor = Color.MintCream; pn3.BorderColor = Color.MintCream;

po1.BorderColor = Color.Red; po2.BorderColor = Color.Red; po3.BorderColor = Color.Red;

node\_n.FillColor = Color.DarkGray;

lb\_n.BackColor = Color.DarkGray;

line\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_o.BorderStyle = System.Drawing.Drawing2D.DashStyle.Solid;

node\_o.BorderColor = Color.Black;

}

}

round++;

}

catch (System.ArgumentOutOfRangeException)

{

runTimer.Enabled = false;

if (MessageBox.Show("ค้นหาโหนด " + TargetBox.Text + " ไม่พบ คุณต้องการค้นหาโหนดอื่นหรือไม่?\nClick No to exit program", "Not found", MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{

start.Text = "START";

Del\_Search();

start.Enabled = true;

Next\_Search();

}

else

{

close();

}

}

if(tlist[0,round-1].Value.ToString()==TargetBox.Text.ToString())

{

runTimer.Enabled = false;

//back.Enabled = false;

//next.Enabled = false;

MessageBox.Show("เจอโหนด " + TargetBox.Text + " ที่ค้นหาแล้ว","Finish",MessageBoxButtons.OK,MessageBoxIcon.Information);

start.Text = "FINISH";

start.Enabled = true;

}

}

//private void TargetBox\_KeyPress(object sender, KeyPressEventArgs e)

//{

// if (e.KeyChar == 'A' || e.KeyChar == 'B' || e.KeyChar == 'C' || e.KeyChar == 'D' || e.KeyChar == 'E' || e.KeyChar == 'F' || e.KeyChar == 'G' || e.KeyChar == 'H' || e.KeyChar == 'I' || e.KeyChar == 'J' || e.KeyChar == 'K' || e.KeyChar == 'L' || e.KeyChar == 'M' || e.KeyChar == 'N' || e.KeyChar == 'O')

// {

// start.Enabled = true;

// }

// else if (e.KeyChar == 'a' || e.KeyChar == 'b' || e.KeyChar == 'c' || e.KeyChar == 'd' || e.KeyChar == 'e' || e.KeyChar == 'f' || e.KeyChar == 'g' || e.KeyChar == 'h' || e.KeyChar == 'i' || e.KeyChar == 'j' || e.KeyChar == 'k' || e.KeyChar == 'l' || e.KeyChar == 'm' || e.KeyChar == 'n' || e.KeyChar == 'o')

// {

// //start.Enabled = false;

// //MessageBox.Show("กรุณากรอกเฉพาะตัวพิมย์ใหญ่เท่านั้น", "Select Node !", MessageBoxButtons.OK, MessageBoxIcon.Warning);

// start.Enabled = true;

// }

// else

// {

// start.Enabled = false;

// MessageBox.Show("กรุณากรอกเฉพาะโหนด A-O เท่านั้น", "Select Node !", MessageBoxButtons.OK, MessageBoxIcon.Warning);

// }

//}

private void run\_Tick(object sender, EventArgs e)

{

Next\_Search();

}

private void lb\_a\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_a.Text;

}

private void lb\_b\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_b.Text;

}

private void lb\_c\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_c.Text;

}

private void lb\_d\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_d.Text;

}

private void lb\_e\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_e.Text;

}

private void lb\_f\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_f.Text;

}

private void lb\_g\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_g.Text;

}

private void lb\_h\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_h.Text;

}

private void lb\_i\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_i.Text;

}

private void lb\_j\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_j.Text;

}

private void lb\_k\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_k.Text;

}

private void lb\_l\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_l.Text;

}

private void lb\_m\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_m.Text;

}

private void lb\_n\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_n.Text;

}

private void lb\_o\_Click(object sender, EventArgs e)

{

TargetBox.Text = lb\_o.Text;

}

private void close()

{

close();

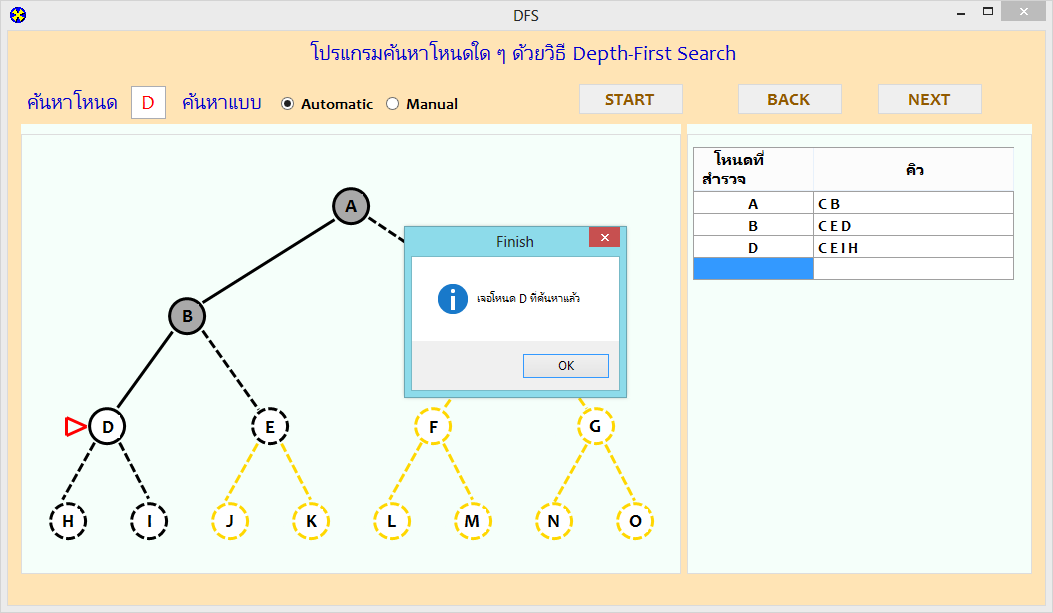
}

}

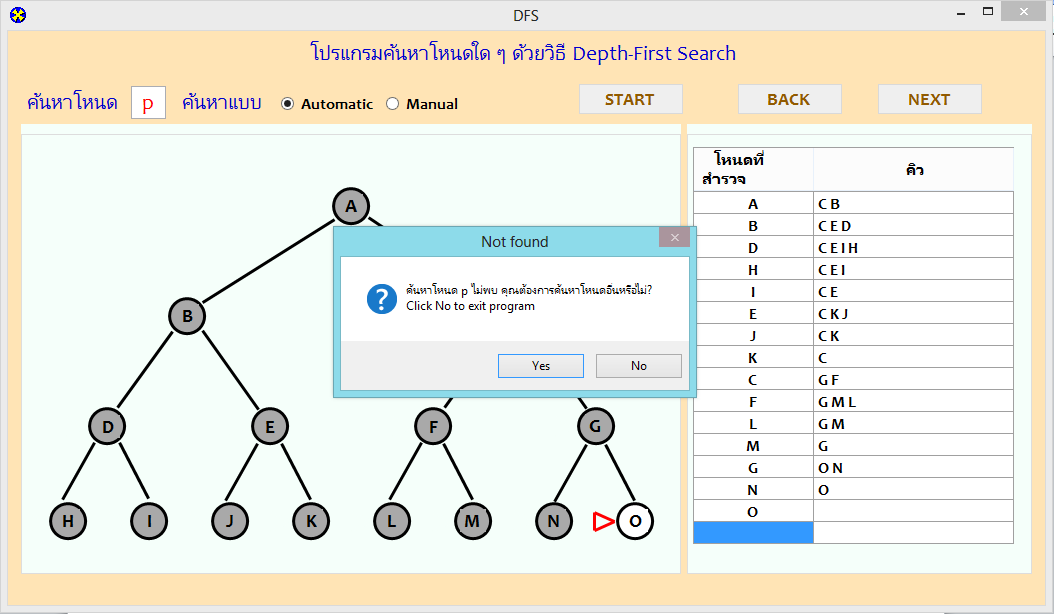
}

**ทดสอบโปรแกรมได้ผลลัพธ์ดังนี้**

**1.ค้นหา D**



**2. ค้นหา p**

****

ค้นหา P ไม่พบเนื่องจาก ตัวแปลที่เก็บไว้ใน Stack ไม่มี จึงแสดงดังรูป

Flow Chart ของโปรแกรม



