

โปรแกรมค้นหาตัวเลข

โดยใช้ ทฤษฎี Depth – First Search

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

โครงสร้างข้อมูลแบบ Stack มีโครงสร้างข้อมูลแบบ Last-in-First-Out คือข้อมูลที่เข้ามาเก็บใน Stack ทีหลังจะเป็นข้อมูลที่ออกจาก Stack ก่อน เช่น การเรียงซ้อนกันของจาน ในการหยิบจานครั้งแรกจะต้องหยิบจานที่อยู่บนสุดก่อน ซึ่งเป็นจานใบล่าสุดที่นำมาวาง เป็นต้น

โครงสร้างข้อมูล Stack

Stack เป็นโครงสร้างข้อมูล ที่มีคุณสมบัติที่ว่าการเพิ่มหรือลบข้อมูลในสแต็กจะกระทำได้ทางเดียว และข้อมูลที่เข้ามาทีหลังจะต้องออกไปก่อน

1. ลักษณะสำคัญของโครงสร้างข้อมูล Stack

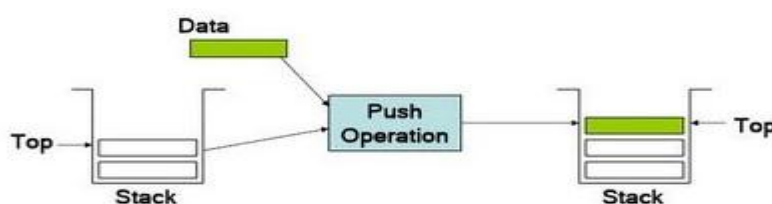
- โครงสร้างข้อมูลเป็นแบบเชิงเส้น ลักษณะโครงสร้างจัดเรียงต่อเนื่องกันไป
- มีลักษณะโครงสร้างที่ไม่ตายตัว สามารถปรับเปลี่ยนจำนวนสมาชิกได้
- นำข้อมูลเข้าและดึงข้อมูลออกได้ สามารถจะทำการ Push และ Pop
- นำข้อมูลเข้าและดึงข้อมูลออกเป็นลำดับ ไม่ข้ามหรือกระโดดไปเอาข้อมูลใดข้อมูลหนึ่งก่อน
- มีการจัดการนำเข้าและดึงข้อมูลในตำแหน่งบนสุด การนำเข้าข้อมูลและการดึงข้อมูลออกต้องทำในตำแหน่งบนสุดเท่านั้น

2. พื้นฐานการดำเนินการของโครงสร้างข้อมูล Stack

Push หรือการนำข้อมูลเข้า

Push คือ การนำข้อมูลใส่ลงไปในสแต็ก เช่น สแต็ก s ต้องการใส่ข้อมูล i ในสแต็กจะได้ Push (s,i) คือ ใส่ข้อมูล i ลงไปที่ท๊อปของสแต็ก s ในการเพิ่มข้อมูลลงในสแต็ก จะต้องทำการตรวจสอบว่าสแต็กเต็มหรือไม่ ถ้าไม่เต็มก็สามารถเพิ่มข้อมูลลงไปในสแต็กได้แล้วปรับตัวชี้ตำแหน่งให้ไปชี้ที่ตำแหน่งข้อมูล

ใหม่ ถ้าสแต็กเต็ม (Stack Overflow) ก็จะไม่สามารถเพิ่มข้อมูลเข้าไปในสแต็กได้อีก

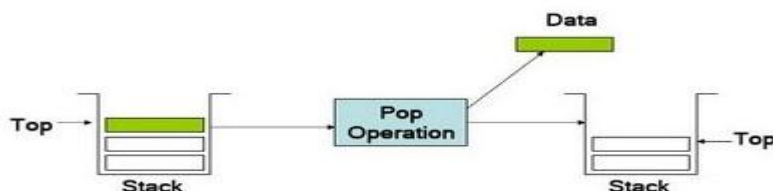


การเพิ่มข้อมูลลงในสแต็ก

Pop หรือการนำออกข้อมูล

Pop คือการนำข้อมูลออกจากส่วนบนสุดของสแต็ก เช่น ต้องการนำข้อมูลออกจาก สแต็ก s ไปไว้ที่ตัวแปร i จะได้ $i = \text{Pop}(s)$ การนำข้อมูลออกจากสแต็ก ถ้าสแต็กมีสมาชิกเพียง 1 ตัวแล้วนำสมาชิกออกจาก

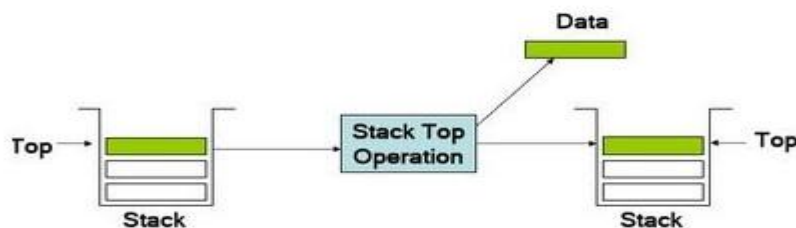
สแต็ก จะเกิดสถานะสแต็กว่าง (Stack Empty) คือไม่มีสมาชิกอยู่ในสแต็กเลยแต่ถ้าไม่มีสมาชิกในสแต็กแล้วทำการ Pop สแต็กจะทำให้เกิดความผิดพลาดที่เรียกว่า Stack Underflow เพราะฉะนั้นก่อนนำข้อมูลออกจากสแต็กจะต้องตรวจสอบก่อนว่าสแต็กว่างหรือเปล่าจึงจะนำข้อมูลออกจากสแต็กได้ และ ปรับตัวชี้ตำแหน่งให้ไปชี้ตำแหน่งของข้อมูลที่ต่อจากข้อมูลที่ถูกลำเอียงไป



การนำข้อมูลออกจากสแต็ก

Top หรือตำแหน่งบนสุด

Stack Top เป็นการคัดลอกข้อมูลที่อยู่บนสุดของสแต็กแต่ไม่ได้นำเอาข้อมูลนั้นออกจากสแต็ก ซึ่ง Top จะเกิดข้อผิดพลาดเช่นเดียวกับการ Pop คือ Stack Underflow เมื่อสแต็กนั้นเกิดการว่าง

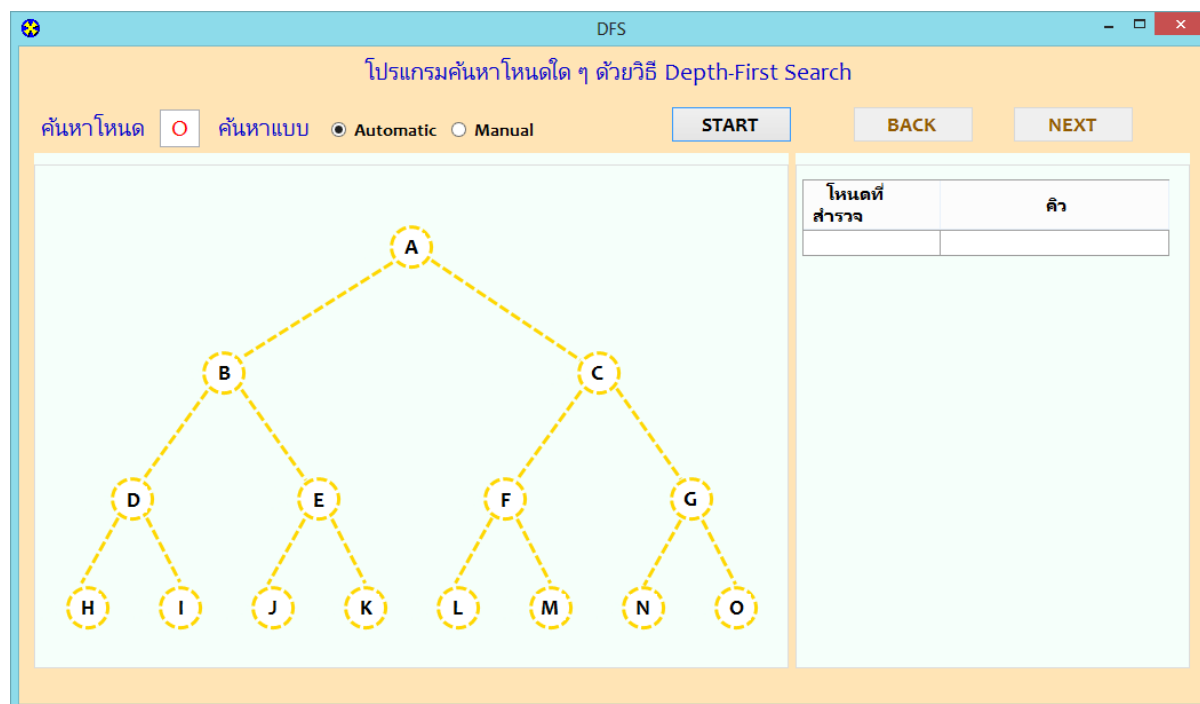


Stack มีพรีออพเพอเรเตอร์และเมธอดพื้นฐานที่ควรรู้ดังนี้

- Push เพิ่มข้อมูลเข้าไปที่ตำแหน่งบนสุดของ Stack
- Pop เอาข้อมูลออกจากตำแหน่งบนสุดของ Stack
- Peek รู้เทิร์นค่าข้อมูลบนสุดของ Stack
- Count จำนวนสมาชิกของ Stack
- Contains ตรวจสอบว่ามีข้อมูลที่ต้องการใน Stack หรือไม่
- To Array เป็นเมธอดที่ใช้ก๊อปปี้ทุกค่าใน Stack ไปสู่อาร์เรย์
- Clear เป็นเมธอดที่ใช้ล้างข้อมูลทุกตัวออกจาก Stack

แสดงแผนการทำงานของโปรแกรม

1.ออกแบบโปรแกรมดังนี้



ตัวอย่างโค้ด

```

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") //กรณีที่มี 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") //คลิกที่ 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") //ดูกรณี 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") //จุดครั้งที่ 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") //จุดครั้งที่ 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") //ถูกวิธีที่ 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") //ถูกวิธีที่ 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") // ក្រចក ជ័រ 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") // ក្រចក ជ័រ 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; //เปลี่ยนสีลูกศร โหนด a ให้เข้ากับสีพื้นหลัง
    pb1.BorderColor = Color.Red; pb2.BorderColor = Color.Red; pb3.BorderColor = Color.Red; //เปลี่ยนสี
ลูกศร โหนด 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("ค้นหาไหนด " + TargetBox.Text + " ไม่พบ คุณต้องการค้นหาไหนดอื่นหรือไม่?\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("เจอไหนด " + TargetBox.Text + " ที่ค้นหา
แล้ว", "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("กรุณากรอกเฉพาะตัวพิมพ์ใหญ่เท่านั้น", "Select Node !", MessageBoxButtons.OK,
MessageBoxIcon.Warning);
//     start.Enabled = true;
// }
// else
// {
//     start.Enabled = false;
//     MessageBox.Show("กรุณากรอกเฉพาะ โหนด 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

DFS

โปรแกรมค้นหาโหนดใด ๆ ด้วยวิธี Depth-First Search

ค้นหาโหนด **D** ค้นหาแบบ ☒ Automatic ☐ Manual

START BACK NEXT

โหนดที่สำรวจ	ค่า
A	CB
B	CED
D	CEIH

2. ค้นหา p

DFS

โปรแกรมค้นหาโหนดใด ๆ ด้วยวิธี Depth-First Search

ค้นหาโหนด **p** ค้นหาแบบ ☒ Automatic ☐ Manual

START BACK NEXT

โหนดที่สำรวจ	ค่า
A	CB
B	CED
D	CEIH
H	CEI
I	CE
E	CKJ
J	CK
K	C
C	GF
F	GML
L	GM
M	G
G	ON
N	O
O	

ค้นหา P ไม่พบเนื่องจาก ตัวแปลที่เก็บไว้ใน Stack ไม่มี จึงแสดง
ดังรูป

Flow Chart ของโปรแกรม

