**เขียนโปรแกรมแสดงขั้นตอนการค้นหาเส้นทางโดยใช้วิธีการค้นหาแบบ Alpha-Bata Pruning?**

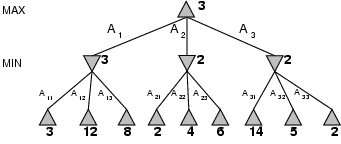


**ทษฎีที่ใช้ในการเขียนโปรแกรม**

**Minimax **เป็นเทคนิคที่ใช้การค้นหาแบบ Depth-first search พิจารณาค่าต่างๆ จากโหนดใบและเลือกค่าที่เหมาะสมส่งผ่านมายังโหนดแม่ จนกระทั่งถึงโหนดราก โดยพิจารณาเลือกค่าจากโหนดใบไม้ และเลือกค่าที่เหมาะสมขึ้นมาบนโหนดพ่อแม่ ทำเช่นนี้ไปจนกว่าโหนดรากจะได้รับค่า เกมที่นำมาพิจารณามักเป็นเกมที่เล่นกัน 2 คน กำหนดให้ผู้เล่นคนหนึ่งเป็น “MAX” (แทนผู้เล่นคือเรา) วัตถุประสงค์ของ MAX คือ การทำคะแนนให้มากที่สุดหรือหาทางชนะ กำหนดให้ผู้เล่นอีกคนหนึ่งเป็น “MIN” วัตถุประสงค์ของ MIN คือ พยายามทำให้ MAX ได้ค่าน้อยที่สุด หรือพยายามทำให้ MAX แพ้นั่นเอง

สัญลักษณ์ของโหนด MAX และ MIN มีลักษณะดังรูป





--------------------------------------------------------------------------------------------------------------------------------------

--------------------------------------------------------------------------------------------------------------------------------------

**การหาค่าของ minimax**

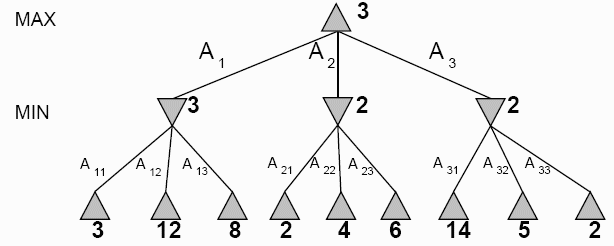
* ถ้าเป็นโหนดใบ ค่าในโหนดคือ Utility function
* ถ้าเป็นโหนดในชั้น MAX ค่าในโหนด คือค่าที่สูงที่สุดของโหนดลูก
* ถ้าเป็นโหนดในชั้น MIN ค่าในโหนดคือค่าที่น้อยที่สุดของโหนดลูก

**Alpha-Beta Pruning** เป็นเทคนิคที่พัฒนาต่อมาจาก Minimax เพื่อตัดเส้นทางบางเส้นทางที่คิดว่าไม่จำเป็นออกให้หมด สำหรับงานที่มีเส้นทางสำรวจเป็นจำนวนมาก และใช้ทรัพยากรในการสำรวจต่ำกว่า Minimax

* Minimax เป็นกระบวนการมุ่งหาเส้นทางจากทุกกิ่ง
* ถ้าเกมมีเส้นทางเยอะจะทาให้เสียเวลาในการค้นหาเส้นทางอย่างมาก 
* Alpha-beta pruning หรือ Alpha-beta cutoff เป็นเทคนิคที่พัฒนาขึ้นมาจาก Minimax เพื่อตัดเส้นทางที่ไม่จาเป็นออกไป ทาให้ประหยัดเวลาในการค้นหาเส้นทาง
* Alpha เป็นค่าที่ดีที่สุดที่นามาช่วยเลือกให้กับโหนด MAX ซึ่งค่าจะไม่สามารถลดลงได้ 
* Beta เป็นค่าที่ดีที่สุดที่นามาช่วยเลือกให้กับโหนด MIN ซึ่งค่าจะไม่สามารถเพิ่มขึ้นได้
* วิธีพิจารณาการตัดกิ่งจะทำได้ เมื่อพิสูจน์ว่าช่วงค่า Alpha จะต้องมากกว่าหรือเท่ากับ Beta จริง

**วิธีการเลือกเส้นทางของ Alpha-beta pruning**

การนำวิธี Alpha-Beta Pruning ใช้กับเกมการเลือกตัวเลขที่ดีที่สุด โดยกำหนดให้แต่ละโหนดต้องบันทึกช่วงของค่าไว้ เพื่อให้ค่า Alpha เลือกค่าที่ดีที่สุดให้โหนด MAX และค่า Beta สามารถเลือกค่าที่ดีที่สุดให้กับโหนด MIN ดังรูป



**Alpha = 3**

**Beta = 3**

**Beta = 2**

**Beta = 2**

--------------------------------------------------------------------------------------------------------------------------------------

--------------------------------------------------------------------------------------------------------------------------------------

* การค้นหาภายใต้โหนด MIN จะหยุดลงต่อเมื่อค่า Beta น้อยกว่าหรือเท่ากับค่า Alpha ของโหนด MAX ที่เป็นพ่อ 
* การค้นหาภายใต้โหนด MAX จะหยุดลงต่อเมื่อค่า Alpha มากกว่าหรือเท่ากับค่า Beta ของโหนด MIN ที่เป็นพ่อ

**โฟลว์ชาร์ตโปรแกรม**

**Start**

**End**

Alpha = min\_set3, beta = min\_set3

Alpha = min\_set1, beta = g Alpha = min\_set1, beta = h Alpha = min\_set1, beta = j

if (g <= set2) else if (h <= set2) else if (j <= set2)

Alpha = min\_set2, beta = min\_set2

Alpha = min\_set1, beta = d Alpha = min\_set1, beta = ee Alpha = min\_set1, beta = f

if (d <= set2) else if (ee <= set2) else if (f <= set2)

min\_set1 = ค่าน้อยที่สุดของ a,b,c

a, b, c, d, ee, f, g, h , j

**Source Code**

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 Alpha\_Bata\_Pruning

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

public int count = 1;

public int count\_step = 1;

public int buff;

private void btnRandom\_Click(object sender, EventArgs e) // random

{

DialogResult result;

if (check())

result = DialogResult.Yes;

else

result = MessageBox.Show("คุณต้องการสุ่มค่าใหม่ทั้งหมดหรือไม่", "สุ่มค่า" , MessageBoxButtons.YesNo, MessageBoxIcon.Information);

if (result == DialogResult.Yes)

{

Random ObjRan = new Random();

nudIn\_1.Value = ObjRan.Next(-99, 99);

nudIn\_2.Value = ObjRan.Next(-99, 99);

nudIn\_3.Value = ObjRan.Next(-99, 99);

nudIn\_4.Value = ObjRan.Next(-99, 99);

nudIn\_5.Value = ObjRan.Next(-99, 99);

nudIn\_6.Value = ObjRan.Next(-99, 99);

nudIn\_7.Value = ObjRan.Next(-99, 99);

nudIn\_8.Value = ObjRan.Next(-99, 99);

nudIn\_9.Value = ObjRan.Next(-99, 99);

}

}

public Boolean check()

{

if (

nudIn\_1.Value == 0 &&

nudIn\_2.Value == 0 &&

nudIn\_3.Value == 0 &&

nudIn\_4.Value == 0 &&

nudIn\_5.Value == 0 &&

nudIn\_6.Value == 0 &&

nudIn\_7.Value == 0 &&

nudIn\_8.Value == 0 &&

nudIn\_9.Value == 0)

return true;

else

return false;

}

public void ABP\_Search() // ค้นหาหลัก

{

if (count\_step == 1)

step\_1();

else if (count\_step == 2)

timer2.Enabled = true;

else if (count\_step == 3)

timer2.Enabled = true;

else if (count\_step == 4)

timer2.Enabled = true;

else if (count\_step == 5)

timer2.Enabled = true;

else if (count\_step == 6)

timer2.Enabled = true;

else if (count\_step == 7)

timer2.Enabled = true;

else if (count\_step == 8)

timer2.Enabled = true;

else if (count\_step == 9)

timer2.Enabled = true;

else if (count\_step == 10)

timer2.Enabled = true;

else if (count\_step == 11)

timer2.Enabled = true;

else if (count\_step == 12)

timer2.Enabled = true;

else if (count\_step == 13)

timer2.Enabled = true;

else if (count\_step == 14)

timer2.Enabled = true;

else if (count\_step == 15)

timer2.Enabled = true;

else if (count\_step == 16)

timer2.Enabled = true;

else if (count\_step == 17)

timer2.Enabled = true;

else if (count\_step == 18)

timer2.Enabled = true;

else if (count\_step == 19)

timer2.Enabled = true;

else if (count\_step == 20)

timer2.Enabled = true;

else if (count\_step == 21)

timer2.Enabled = true;

else if (count\_step == 22)

timer2.Enabled = true;

else if (count\_step == 23)

timer2.Enabled = true;

else if (count\_step == 24)

timer2.Enabled = true;

else if (count\_step == 25)

timer2.Enabled = true;

else if (count\_step == 26)

timer2.Enabled = true;

else if (count\_step == 27)

timer2.Enabled = true;

else if (count\_step == 28)

timer2.Enabled = true;

else if (count\_step == 29)

timer2.Enabled = true;

else if (count\_step == 30)

timer2.Enabled = true;

else if (count\_step == 31)

timer2.Enabled = true;

else if (count\_step == 32)

timer2.Enabled = true;

else if (count\_step == 33)

timer2.Enabled = true;

else if (count\_step == 34)

timer2.Enabled = true;

else if (count\_step == 35)

timer2.Enabled = true;

else if (count\_step == 99)

timer2.Enabled = true;

}

/// <1>

/// ////////////////

/// </summary>

public void step\_1() // โชว์ค่าจาก input

{

label1.Text = nudIn\_1.Value.ToString();

label2.Text = nudIn\_2.Value.ToString();

label3.Text = nudIn\_3.Value.ToString();

label4.Text = nudIn\_4.Value.ToString();

label5.Text = nudIn\_5.Value.ToString();

label6.Text = nudIn\_6.Value.ToString();

label7.Text = nudIn\_7.Value.ToString();

label8.Text = nudIn\_8.Value.ToString();

label9.Text = nudIn\_9.Value.ToString();

label1.Visible = true;

label2.Visible = true;

label3.Visible = true;

label4.Visible = true;

label5.Visible = true;

label6.Visible = true;

label7.Visible = true;

label8.Visible = true;

label9.Visible = true;

}

public void step\_2()

{

if (count == 1)

ptbGreen\_13.Visible = true;

else if (count == 2)

{

ptbGreen\_13.Visible = false;

labRoot.Visible = true;

timer2.Enabled = false;

}

count++;

}

public void step\_3()

{

if (count == 3)

ptbGreen\_10.Visible = true;

else if (count == 4)

{

ptbGreen\_10.Visible = false;

labMin\_1.Visible = true;

timer2.Enabled = false;

}

count++;

}

public void step\_4()

{

if (count == 5)

{

ptbGreen\_1.Visible = true;

label1.BackColor = Color.Lime;

}

else if (count == 6)

{

ptbGreen\_1.Visible = false;

ptbRad\_1.Visible = true;

label1.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_5()

{

if (count == 7)

{

ptbGreen\_10.Visible = true;

}

else if (count == 8)

{

ptbGreen\_10.Visible = false;

label10.Visible = true;

ptbRad\_10.Visible = true;

label10.BackColor = Color.Red;

label10.Text = label1.Text;

labMin\_1.Text = "(-infinite , " + label1.Text + " ]";

timer2.Enabled = false;

}

count++;

}

public void step\_6()

{

if (count == 9)

{

label2.BackColor = Color.Lime;

ptbGreen\_2.Visible = true;

}

else if (count == 10)

{

ptbGreen\_2.Visible = false;

ptbRad\_2.Visible = true;

label2.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_7()

{

if (count == 11)

{

ptbRad\_10.Visible = false;

ptbGreen\_10.Visible = true;

}

else if (count == 12)

{

if (Convert.ToInt32(label2.Text) <= Convert.ToInt32(label10.Text))

{

label10.Text = label2.Text;

labMin\_1.Text = "(-infinite , " + label2.Text + " ]";

}

ptbGreen\_10.Visible = false;

label10.Visible = true;

ptbRad\_10.Visible = true;

label10.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_8()

{

if (count == 13)

{

label3.BackColor = Color.Lime;

ptbGreen\_3.Visible = true;

}

else if (count == 14)

{

ptbGreen\_3.Visible = false;

ptbRad\_3.Visible = true;

label3.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_9()

{

if (count == 15)

{

ptbRad\_10.Visible = false;

ptbGreen\_10.Visible = true;

}

else if (count == 16)

{

if (Convert.ToInt32(label3.Text) <= Convert.ToInt32(label10.Text))

{

label10.Text = label3.Text;

labMin\_1.Text = "[ " + label10.Text + " , " + label10.Text + " ]";

}

labMin\_1.Text = "[ " + label10.Text + " , " + label10.Text + " ]";

ptbGreen\_10.Visible = false;

label10.Visible = true;

ptbRad\_10.Visible = true;

label10.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_10()

{

if (count == 17)

{

ptbRad\_13.Visible = false;

ptbGreen\_13.Visible = true;

}

else if (count == 18)

{

buff = Convert.ToInt32(label10.Text);

label13.Text = buff.ToString();

labRoot.Text = "[ " + label13.Text + " , +infinite )";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

} // ขึ้นหา root

/// <2>

/// //////////////////////

/// </summary>

public void step\_11()

{

if (count == 19)

{

ptbGreen\_11.Visible = true;

}

else if (count == 20)

{

labMin\_2.Visible = true;

ptbGreen\_11.Visible = false;

timer2.Enabled = false;

}

count++;

}

public void step\_12()

{

if (count == 21)

{

ptbGreen\_4.Visible = true;

label4.BackColor = Color.Lime;

}

else if (count == 22)

{

ptbGreen\_4.Visible = false;

ptbRad\_4.Visible = true;

label4.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_13()

{

if (count == 23)

{

ptbGreen\_11.Visible = true;

}

else if (count == 24)

{

ptbGreen\_11.Visible = false;

label11.Visible = true;

ptbRad\_11.Visible = true;

label11.BackColor = Color.Red;

label11.Text = label4.Text;

labMin\_2.Text = "(-infinite , "+ label11.Text + " ]" ;

timer2.Enabled = false;

}

count++;

}

public void step\_14() // ขึ้นหา root

{

if (count == 25)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 26)

{

if (buff >= Convert.ToInt32(label11.Text))// cut off

{

labRoot.Text = "[ " + label11.Text + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

labA\_B.Visible = true;

count\_step = 23;

count = 42;

}

else

{

//.Text = label11.Text;

labRoot.Text = "[ " + label13.Text + " , " + label11.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

timer2.Enabled = false;

}

count++;

}

public void step\_15() // lab 11 กระพริบ ไม่เปลี่ยนค่า

{

if (count == 27)

{

ptbGreen\_11.Visible = true;

label11.BackColor = Color.Lime;

}

else if (count == 28)

{

ptbGreen\_11.Visible = false;

label11.Visible = true;

ptbRad\_11.Visible = true;

label11.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_16() // lab 5 กระพริบ

{

if (count == 29)

{

ptbGreen\_5.Visible = true;

label5.BackColor = Color.Lime;

}

else if (count == 30)

{

ptbGreen\_5.Visible = false;

label5.Visible = true;

ptbRad\_5.Visible = true;

label5.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_17() // lab 11 กระพริบ เปลี่ยนค่า

{

if (count == 31)

{

ptbRad\_11.Visible = false;

ptbGreen\_11.Visible = true;

}

else if (count == 32)

{

if (Convert.ToInt32(label5.Text) <= Convert.ToInt32(label11.Text))

{

label11.Text = label5.Text;

}

labMin\_2.Text = "(-infinite , " + label11.Text + " ]";

ptbGreen\_11.Visible = false;

label11.Visible = true;

ptbRad\_11.Visible = true;

label11.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_18() // ขึ้นหา root

{

if (count == 33)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 34)

{

if (buff >= Convert.ToInt32(label11.Text))// cut off

{

buff = Convert.ToInt32(label11.Text);

labRoot.Text = "[ " + buff.ToString() + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

labA\_B.Visible = true;

count\_step = 23;

count = 42;

}

else

{

//label13.Text = label11.Text;

labRoot.Text = "[ " + label13.Text + " , " + label11.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

timer2.Enabled = false;

}

count++;

}

public void step\_19() // lab 11 กระพริบ ไม่เปลี่ยนค่า

{

if (count == 35)

{

ptbGreen\_11.Visible = true;

label11.BackColor = Color.Lime;

}

else if (count == 36)

{

ptbGreen\_11.Visible = false;

label11.Visible = true;

ptbRad\_11.Visible = true;

label11.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_20() // lab 6 กระพริบ

{

if (count == 37)

{

ptbGreen\_6.Visible = true;

label6.BackColor = Color.Lime;

}

else if (count == 38)

{

ptbGreen\_6.Visible = false;

label6.Visible = true;

ptbRad\_6.Visible = true;

label6.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_21() // lab 11 กระพริบ เปลี่ยนค่า

{

if (count == 39)

{

ptbRad\_11.Visible = false;

ptbGreen\_11.Visible = true;

}

else if (count == 40)

{

if (Convert.ToInt32(label6.Text) <= Convert.ToInt32(label11.Text))

{

label11.Text = label6.Text;

}

labMin\_2.Text = "[ " + label11.Text + " , " + label11.Text + " ]";

ptbGreen\_11.Visible = false;

label11.Visible = true;

ptbRad\_11.Visible = true;

label11.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_22() // ขึ้นหา root

{

if (count == 41)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 42)

{

if (buff >= Convert.ToInt32(label11.Text))// cut off

{

buff = Convert.ToInt32(label11.Text);

labRoot.Text = "[ " + buff.ToString() + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

//labA\_B.Visible = true;

count\_step = 23;

count = 42;

}

else

{

labRoot.Text = "[ " + label11.Text + " , " + label13.Text + " ]";

label13.Text = label11.Text;

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

timer2.Enabled = false;

}

count++;

}

/// <3>

/// ////////////////////////////

/// </summary>

public void step\_23()

{

if (count == 43)

{

ptbGreen\_12.Visible = true;

}

else if (count == 44)

{

labMin\_3.Visible = true;

ptbGreen\_12.Visible = false;

timer2.Enabled = false;

}

count++;

}

public void step\_24()

{

if (count == 45)

{

ptbGreen\_7.Visible = true;

label7.BackColor = Color.Lime;

}

else if (count == 46)

{

ptbGreen\_7.Visible = false;

ptbRad\_7.Visible = true;

label7.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_25()

{

if (count == 47)

{

ptbRad\_12.Visible = false;

ptbGreen\_12.Visible = true;

label12.BackColor = Color.Lime;

}

else if (count == 48)

{

ptbGreen\_12.Visible = false;

label12.Visible = true;

ptbRad\_12.Visible = true;

label12.BackColor = Color.Red;

label12.Text = label7.Text;

labMin\_3.Text = "(-infinite , " + label12.Text + " ]";

timer2.Enabled = false;

}

count++;

}

public void step\_26() // ขึ้นหา root

{

buff = Convert.ToInt32(label13.Text);

if (count == 49)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 50)

{

if (buff >= Convert.ToInt32(label12.Text))// cut off

{

labRoot.Text = "[ " + label12.Text + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

labA\_B2.Visible = true;

count\_step = 99;

count = 66;

}

else

{

labRoot.Text = "[ " + label13.Text + " , " + label12.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

timer2.Enabled = false;

}

count++;

}

public void step\_27() // lab 12 กระพริบ ไม่เปลี่ยนค่า

{

if (count == 51)

{

ptbGreen\_12.Visible = true;

label12.BackColor = Color.Lime;

}

else if (count == 52)

{

ptbGreen\_12.Visible = false;

label12.Visible = true;

ptbRad\_12.Visible = true;

label12.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_28() // lab 8 กระพริบ

{

if (count == 53)

{

ptbGreen\_8.Visible = true;

label8.BackColor = Color.Lime;

}

else if (count == 54)

{

ptbGreen\_8.Visible = false;

label8.Visible = true;

ptbRad\_8.Visible = true;

label8.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_29() // lab 12 กระพริบ เปลี่ยนค่า

{

if (count == 55)

{

ptbRad\_12.Visible = false;

ptbGreen\_12.Visible = true;

}

else if (count == 56)

{

if (Convert.ToInt32(label8.Text) <= Convert.ToInt32(label12.Text))

{

label12.Text = label8.Text;

labMin\_3.Text = "(-infinite , " + label12.Text + " ]";

}

ptbGreen\_12.Visible = false;

label12.Visible = true;

ptbRad\_12.Visible = true;

label12.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_30() // ขึ้นหา root

{

if (count == 57)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 58)

{

if (buff >= Convert.ToInt32(label12.Text))// cut off

{

//buff = Convert.ToInt32(label11.Text);

labRoot.Text = "[ " + label12.Text + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

labA\_B2.Visible = true;

count\_step = 99;

count = 66;

}

else

{

//label13.Text = label11.Text;

labRoot.Text = "[ " + label13.Text + " , " + label12.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

timer2.Enabled = false;

}

count++;

}

public void step\_31() // lab 12 กระพริบ ไม่เปลี่ยนค่า

{

if (count == 59)

{

ptbGreen\_12.Visible = true;

label12.BackColor = Color.Lime;

}

else if (count == 60)

{

ptbGreen\_12.Visible = false;

label12.Visible = true;

ptbRad\_12.Visible = true;

label12.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_32() // lab 9 กระพริบ

{

if (count == 61)

{

ptbGreen\_9.Visible = true;

label9.BackColor = Color.Lime;

}

else if (count == 62)

{

ptbGreen\_9.Visible = false;

label9.Visible = true;

ptbRad\_9.Visible = true;

label9.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_33() // lab 12 กระพริบ เปลี่ยนค่า

{

if (count == 63)

{

ptbRad\_12.Visible = false;

ptbGreen\_12.Visible = true;

}

else if (count == 64)

{

if (Convert.ToInt32(label9.Text) <= Convert.ToInt32(label12.Text))

{

label12.Text = label9.Text;

labMin\_3.Text = "[ " + label12.Text + " , " + label12.Text + " ]";

}

labMin\_3.Text = "[ " + label12.Text + " , " + label12.Text + " ]";

ptbGreen\_12.Visible = false;

label12.Visible = true;

ptbRad\_12.Visible = true;

label12.BackColor = Color.Red;

timer2.Enabled = false;

}

count++;

}

public void step\_34() // ขึ้นหา root

{

if (count == 65)

{

ptbGreen\_13.Visible = true;

label13.BackColor = Color.Lime;

label13.Visible = false;

}

else if (count == 66)

{

if (buff >= Convert.ToInt32(label12.Text))// cut off

{

//buff = Convert.ToInt32(label12.Text);

labRoot.Text = "[ " + label12.Text + " , " + label13.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

labA\_B2.Visible = true;

}

else

{

labRoot.Text = "[ " + label13.Text + " , " + label12.Text + " ]";

ptbGreen\_13.Visible = false;

label13.Visible = true;

ptbRad\_13.Visible = true;

label13.BackColor = Color.Red;

}

count\_step = 99;

count = 66;

timer2.Enabled = false;

}

count++;

}

public void step\_99() // end 13 กระพริบ

{

if (count == 67)

{

ptbGreen\_13.Visible = false;

ptbRad\_13.Visible = true;

label13.Visible = false;

}

else if (count == 68)

{

timer1.Enabled = false;

timer2.Enabled = false;

labFin.Text = check\_Num();

labRoot.Text = "[ " + label13.Text + " , " + label13.Text + " ]";

ptbRad\_13.Visible = false;

labFin.Visible = true;

ptbGreen\_13.Visible = true;

labFin.BackColor = Color.Lime;

btnStart.Enabled = false;

// btnClear.Text = "Clear";

MessageBox.Show("การค้นหาเสร็จสิ้น", "การค้นหา",MessageBoxButtons.OK,MessageBoxIcon.Information);

}

count++;

}

public string check\_Num()

{

string strOut = label10.Text;

if (Convert.ToInt32(strOut) < Convert.ToInt32(label11.Text))

strOut = label11.Text;

if (Convert.ToInt32(strOut) < Convert.ToInt32(label12.Text))

strOut = label12.Text;

return strOut;

}

private void timer2\_Tick(object sender, EventArgs e)

{

if (count\_step == 3)

step\_2();

else if (count\_step == 4)

step\_3();

else if (count\_step == 5)

step\_4();

else if (count\_step == 6)

step\_5();

else if (count\_step == 7)

step\_6();

else if (count\_step == 8)

step\_7();

else if (count\_step == 9)

step\_8();

else if (count\_step == 10)

step\_9();

else if (count\_step == 11)

step\_10();

else if (count\_step == 12)

step\_11();

else if (count\_step == 13)

step\_12();

else if (count\_step == 14)

step\_13();

else if (count\_step == 15)

step\_14();

else if (count\_step == 16)

step\_15();

else if (count\_step == 17)

step\_16();

else if (count\_step == 18)

step\_17();

else if (count\_step == 19)

step\_18();

else if (count\_step == 20)

step\_19();

else if (count\_step == 21)

step\_20();

else if (count\_step == 22)

step\_21();

else if (count\_step == 23)

step\_22();

else if (count\_step == 24)

step\_23();

else if (count\_step == 25)

step\_24();

else if (count\_step == 26)

step\_25();

else if (count\_step == 27)

step\_26();

else if (count\_step == 28)

step\_27();

else if (count\_step == 29)

step\_28();

else if (count\_step == 30)

step\_29();

else if (count\_step == 31)

step\_30();

else if (count\_step == 32)

step\_31();

else if (count\_step == 33)

step\_32();

else if (count\_step == 34)

step\_33();

else if (count\_step == 35)

step\_34();

else if (count\_step == 100)

step\_99();

if (chbAuto.Checked == true && count % 2 != 0)

btnStart.Enabled = true;

else if (chbAuto.Checked == true && count % 2 == 0)

btnStart.Enabled = false;

}

private void btnStart\_Click(object sender, EventArgs e) // ปุ่มเริ่มต้น

{

btnClear.Text = "Stop";

if (chbAuto.Checked == true)

{

timer1\_Tick(sender, e);

btnStart.Text = "Next";

visi();

}

else

{

if (btnStart.Text == "Pause")

{

timer1.Enabled = false;

btnStart.Text = "Play";

}

else

{

timer1.Enabled = true;

visi();

btnStart.Text = "Pause";

}

}

}

public void visi()

{

chbAuto.Enabled = false;

btnRandom.Enabled = false;

chbAuto.Enabled = false;

btnRandom.Enabled = false;

nudIn\_1.Enabled = false;

nudIn\_2.Enabled = false;

nudIn\_3.Enabled = false;

nudIn\_4.Enabled = false;

nudIn\_5.Enabled = false;

nudIn\_6.Enabled = false;

nudIn\_7.Enabled = false;

nudIn\_8.Enabled = false;

nudIn\_9.Enabled = false;

}

private void timer1\_Tick(object sender, EventArgs e) // นาฬิกาทำงาน

{

ABP\_Search();

count\_step++;

}

private void btnClear\_Click(object sender, EventArgs e) // clear

{

if (btnClear.Text == "Clear") //clear

{

DialogResult dig = MessageBox.Show("คุณต้องล้างค่าทั้งหมดหรือไม่", "Clear", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

if (dig == DialogResult.Yes)

{

F\_clear();

F\_stop();

}

}

else // stop

{

btnClear.Text = "Clear";

F\_stop();

}

}

public void F\_clear()

{

timer1.Enabled = false;

timer2.Enabled = false;

nudIn\_1.Value = 0;

nudIn\_2.Value = 0;

nudIn\_3.Value = 0;

nudIn\_4.Value = 0;

nudIn\_5.Value = 0;

nudIn\_6.Value = 0;

nudIn\_7.Value = 0;

nudIn\_8.Value = 0;

nudIn\_9.Value = 0;

}

public void F\_stop()

{

timer1.Enabled = false;

timer2.Enabled = false;

ptbGreen\_1.Visible = false;

ptbGreen\_2.Visible = false;

ptbGreen\_3.Visible = false;

ptbGreen\_4.Visible = false;

ptbGreen\_5.Visible = false;

ptbGreen\_6.Visible = false;

ptbGreen\_7.Visible = false;

ptbGreen\_8.Visible = false;

ptbGreen\_9.Visible = false;

ptbGreen\_10.Visible = false;

ptbGreen\_11.Visible = false;

ptbGreen\_12.Visible = false;

ptbGreen\_13.Visible = false;

ptbRad\_1.Visible = false;

ptbRad\_2.Visible = false;

ptbRad\_3.Visible = false;

ptbRad\_4.Visible = false;

ptbRad\_5.Visible = false;

ptbRad\_6.Visible = false;

ptbRad\_7.Visible = false;

ptbRad\_8.Visible = false;

ptbRad\_9.Visible = false;

ptbRad\_10.Visible = false;

ptbRad\_11.Visible = false;

ptbRad\_12.Visible = false;

ptbRad\_13.Visible = false;

label1.Visible = false;

label2.Visible = false;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

label7.Visible = false;

label8.Visible = false;

label9.Visible = false;

label10.Visible = false;

label11.Visible = false;

label12.Visible = false;

label13.Visible = false;

labFin.Visible = false;

labMin\_1.Visible = false;

labMin\_2.Visible = false;

labMin\_3.Visible = false;

labA\_B.Visible = false;

labA\_B2.Visible = false;

labRoot.Visible = false;

chbAuto.Enabled = true;

btnRandom.Enabled = true;

btnStart.Enabled = true;

nudIn\_1.Enabled = true;

nudIn\_2.Enabled = true;

nudIn\_3.Enabled = true;

nudIn\_4.Enabled = true;

nudIn\_5.Enabled = true;

nudIn\_6.Enabled = true;

nudIn\_7.Enabled = true;

nudIn\_8.Enabled = true;

nudIn\_9.Enabled = true;

count = 1;

count\_step = 1;

label1.BackColor = Color.Transparent;

label2.BackColor = Color.Transparent;

label3.BackColor = Color.Transparent;

label4.BackColor = Color.Transparent;

label5.BackColor = Color.Transparent;

label6.BackColor = Color.Transparent;

label7.BackColor = Color.Transparent;

label8.BackColor = Color.Transparent;

label9.BackColor = Color.Transparent;

btnStart.Text = "Start";

labRoot.Text = "(-infinite , +infinite)";

labMin\_1.Text = "(-infinite , +infinite)";

labMin\_2.Text = "(-infinite , +infinite)";

labMin\_3.Text = "(-infinite , +infinite)";

}

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

{

MessageBox.Show(about(), "About", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

public string about()

{

string str = "โปรแกรมแสดงขั้นตอนการค้นหาเส้นทางโดยใช้วิธีการค้นหาแบบ Alpha-Beta Pruning\n Version 1.0.1\n\nDeveloped by EvilEdition Studio";

return str;

}

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

{

MessageBox.Show(what(), "Alpha-Beta Pruning",MessageBoxButtons.OK, MessageBoxIcon.Information);

}

public string what()

{

string str = " Alpha-Beta นั้นเป็นอัลกอที่เหมาะกับ AI จำพวกหมากกระดาน เป็นอัลกอคิดคิดต่อเนื่องมาจาก MiniMax โดยมีแนวความคิดในการแตก Tree ทางเดินที่เป็นไปได้ทั้งหมดออกมา แล้วหาเส้นทางที่ดีที่สุดโดยมองล่วงหน้าไป (จะกี่ชั้นก็แล้วแต่) แบ่งเป็นตาเดินของเราและตาของฝ่ายตรงข้ามสลับๆไป จนถึงจำนวนชั้นล่วงหน้าที่กำหนด แล้วคิดคะแนนในความลึกชั้นสุดท้ายแล้ว return คะแนน กลับออกมาแล้วเอาคะแนนมาคิดหาทางเดินที่ควรจะเดินโดยAssumeว่า ในตาของฝ่ายตรงข้ามๆจะเลือกตาเดินที่แย่ที่สุดของเรา(ก็คือตาที่ดีที่สุดของฝ่ายตรงข้าม)และในความลึกที่เป็นตาของเรา เราก็จะเลือกตาที่ดีที่สุดของเราจนฟังก์ชั้น return กลับมาหมดเราก็จะได้ตาที่ดีที่สุดออกมา (ซึ่งขึ้นอยู่กับการตีคะแนนในชั้นสุดท้าย และ จำนวนชั้นที่มองล่วงหน้าเข้าไป)อันนี้คือคราวๆของ Minimax โดยที่ Alpha-Beta จะเป็นส่วนเสริมของMinimaxโดย Alpha-Beta จะตัดตาเดินที่ไม่จำเป็นต้องคิดออกไปตาเดินที่ว่าคือตาเดินในลักษณะที่ว่าเป็นตาเดินที่ยังไงซะ ในความลึกในระดับชั้นบนถัดไป ยังไงก็ไม่เลือกไปคิดอยู่แล้วคือมันเป็นตาที่ไม่ดีในความลึกระดับชั้นบนถัดไป";

return str;

}

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

{

MessageBox.Show(what\_(), "คำแนะนำในการใช้", MessageBoxButtons.OK, MessageBoxIcon.Information);

}

public string what\_()

{

string str = " คลิกเครื่องหมายถูกที่ช่อง \"Step by Step\" เพื่อเลือกการทำงานแบบขั้นต่อขั้น หรือคลิกเครื่องหมายถูกออกเพื่อเลือกการทำงานแบบอัตโนมัติ";

str += "\n หลังจาที่กดปุ่ม \"Start\" สามารถกดปุ่มนั้นอีกครั้งเพื่อหยุดการทำงานชั่วคราวในโหมดอัตโนมัติ หรือทำงานในขั้นต่อไปในโหมด Step by Step";

str += "\n ในระหว่างระบบกำลังงานสามารถกดปุ่ม \"Stop\" เพื่อหยุดการทำงานทั้งหมด ซึ่งสามารถกดปุ่ม \"Start\" เพื่อเริ่มการทำงานเดิมได้อีกครั้ง และถ้าหากระบบไม่มีการทำงาน สามารถกดปุ่ม \"Clear\" เพื่อล้างค่าทั้งหมดให้ไปอยู่ในสถานะเริ่มตั้น";

str += "\n กดปุ่ม \"Random\" เพื่อสุ่มค่าตัวเลขที่ป้อนเข้าไปในการค้นหา ถ้าหากก่อนหน้านั้นมีการใส่หรือสุ่มตัวเลขไว้แล้ว(ตัวเลขที่ Input ไม่เท่ากับ 0)ระบบจะถามเพื่อยืนยันการสุ่มตัวเลข";

return str;

}

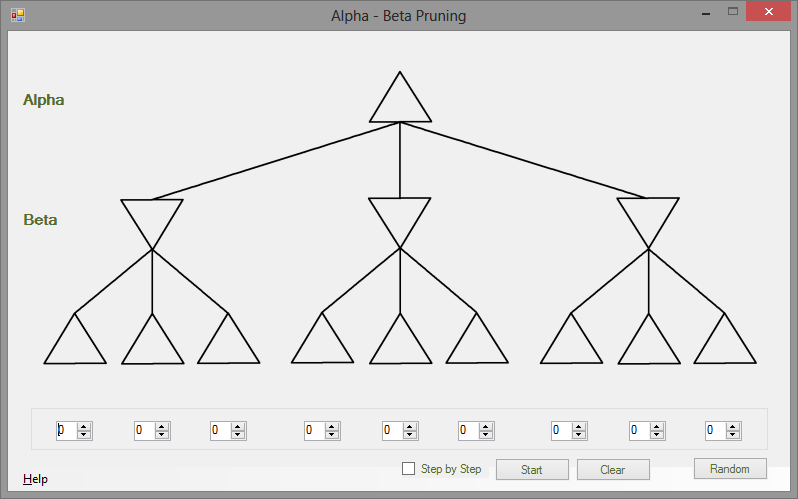
}

}

**การทำงานของโปรแกรม**

หน้าจอเริ่มต้น

ส่วนแสดงผลการทำงานทั้งหมดของโปรแกรม



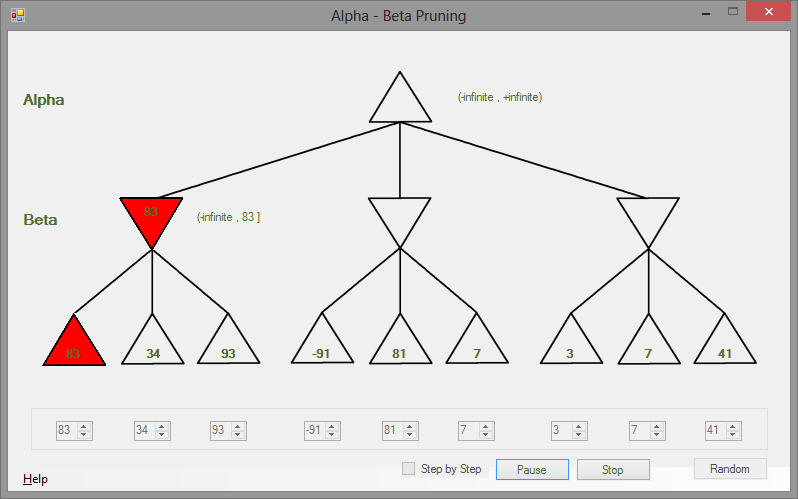
เลือกการทำงานเป็นแบบ Auto หรือ Step

สามารถใส่ตัวเลขที่ต้องการในแต่ละตำแหน่ง

สุ่มตัวเลขที่จะป้อนไปยัง Input

ขณะโปรแกรมทำงาน

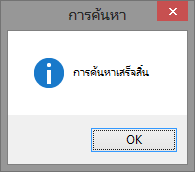
โปรแกรมทำการค้นหาทีละโหนด



หยุดการทำงานทั้งหมด แต่ยังสามารถเริ่มการทำงานใหม่ได้

หยุดชั่วคราว

การทำงานเสร็จสิ้น



โหนดที่ได้จากการค้นหา

