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 Visual\_Code

{

public partial class Nqueen : Form

{

public Nqueen()

{

InitializeComponent();

}

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

{

}

private void button1\_Click(object sender, EventArgs e)

{

if (panel1.Visible == false)

{

panel1.Visible = true;

}

if (panel2.Visible == false)

{

panel2.Visible = true;

}

if (panel3.Visible == false)

{

panel3.Visible = true;

}

if (panel4.Visible == false)

{

panel4.Visible = true;

}

if (panel5.Visible == false)

{

panel5.Visible = true;

}

if (panel6.Visible == false)

{

panel6.Visible = true;

}

if (panel7.Visible == false)

{

panel7.Visible = true;

}

if (panel8.Visible == false)

{

panel8.Visible = true;

}

if (panel9.Visible == false)

{

panel9.Visible = true;

}

if (panel10.Visible == false)

{

panel10.Visible = true;

}

if (panel11.Visible == false)

{

panel11.Visible = true;

}

if (panel12.Visible == false)

{

panel12.Visible = true;

}

if (panel13.Visible == false)

{

panel13.Visible = true;

}

if (panel14.Visible == false)

{

panel14.Visible = true;

}

if (5== 5)

{

int L = 5;

int x = 20, y = 100;

S = L;

int[,] arr = new int[L, L];

if (!theBoardSolver(arr, 0))

{ }

for (int i = 0; i < L; i++)

{

for (int j = 0; j < L; j++)

{

Panel pn = new Panel();

pn.Width = 60;

pn.Height = 50;

pn.Left = x;

pn.Top = y;

pn.Name = pn + "" + i;

pn.BackColor = Color.Green;

pn.BackColor = Color.Green;

this.Controls.Add(pn);

x += 75;

if (arr[i, j] == 1)

{

pn.BackColor = Color.Blue;

}

if (arr[i,j]==0)

{

pn.BackColor = Color.Green;

}

if (arr[i, j] == 2)

{

pn.BackColor = Color.Red;

}

}

x = 20;

y += 65;

}

for (int i = 0; i < L; i++)

{

for (int j = 0; j < L; j++)

{

Panel pn2 = new Panel();

pn2.Width = 60;

pn2.Height = 50;

pn2.Left = x;

pn2.Top = y;

pn2.Name = pn2 + "" + i;

pn2.BackColor = Color.Green;

this.Controls.Add(pn2);

if (arr[i, j] == 3)

{

pn2.BackColor = Color.Red;

}

if (arr[i, j] == 4)

{

pn2.BackColor = Color.Green;

}

if (arr[i, j] == 5)

{

pn2.BackColor = Color.Blue;

}

}

}

}

}

static int S;

static Boolean colorss(int[,] board, int row, int col)

{

int i, j;

for (i = 0; i < col; i++)

{

if (board[row, i] == 1)

return false;

}

for (i = row, j = col; i >= 0 && j >= 0; i--, j--)

{

if (board[i, j] == 1) return false;

}

for (i = row, j = col; j >= 0 && i < S; i++, j--)

{

if (board[i, j] == 1) return false;

}

//--------------------------------------

//-------------------------------

return true;

}

static Boolean theBoardSolver(int[,] board, int col)

{

if (col >= S)

return true;

for (int i = 0; i < S; i++)

{

if (colorss(board, i, col))

{

board[i, col] = 1;

if (theBoardSolver(board, col + 1))

return true;

}

if (colorss(board, i, col))

{

board[i, col] = 2;

if (theBoardSolver(board, col + 1))

return true;

}

}

return false;

}

static Boolean Coloor(int[,] board, int col)

{

if (col >= S)

return true;

for (int i = 0; i < S; i++)

{

if (colorss(board, i, col))

{

board[i, col] = 2;

if (theBoardSolver(board, col + 1))

return true;

}

}

return false;

}

}

}



