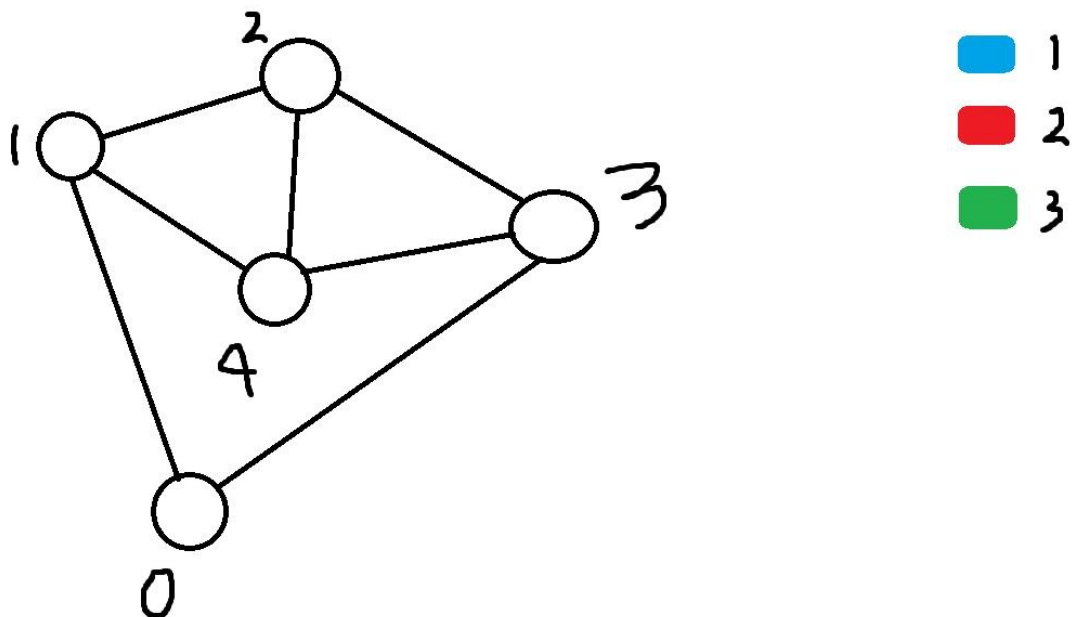


گزارش کار:

### رنگ آمیزی گراف با روش (ABT) Backtracking

در این برنامه قصد داریم رئوس یک گراف را طوری رنگ آمیزی کنیم که هیچ رئوس مجاوری (دو رأس که با هم ارتباط دارند) دارای رنگ یکسان نباشند. فرض کنید گراف زیر که دارای ۵ رأس می باشد را می خواهیم با ۳ رنگ آبی، قرمز و سبز رنگ آمیزی کنیم.



پس از اجرای برنامه گراف مورد نظر باید توسط کاربر معرفی شود که مراحل آن در ادامه آمده است:

(۱) تعیین تعداد رئوس

```
--- ABT graph coloring problem ---  
Vertices count:  
5
```

(۲) تعیین ارتباط میان رئوس. پس از انتخاب تعداد رئوس گراف برنامه به ترتیب به ازای هر دو رأس وجود ارتباط میان آن دو را سوال می کند. در صورتی که میان دو رأس ارتباط وجود داشته باشد عدد ۱ و در غیر اینصورت عدد ۰ وارد می شود. توجه داشته باشید که هر رأس با خودش ارتباط ندارد لذا سوال ارتباط هر رأس با خودش پرسیده نمی شود.

```

--- ABT graph coloring problem ---
Vertices count:
5
If vertex i has link with vertex j enter 1 otherwise 0:
0 -> 1 ?
1
0 -> 2 ?
0
0 -> 3 ?
1
0 -> 4 ?
0
1 -> 0 ?
1
1 -> 2 ?
1
1 -> 3 ?
0
1 -> 4 ?
1
2 -> 0 ?
0
2 -> 1 ?
1
2 -> 3 ?
1
2 -> 4 ?
1
3 -> 0 ?
0
3 -> 1 ?
0
3 -> 2 ?
1
3 -> 4 ?
1
4 -> 0 ?
0
4 -> 1 ?
1
4 -> 2 ?
1
4 -> 3 ?
1
1

```

(۳) نمایش ماتریس گراف. پس از وارد کردن ارتباطات میان رئوس ماتریس گراف توسط برنامه ساخته شده و نمایش داده می شود.

```

3 -> 2 ?
1
3 -> 4 ?
1
4 -> 0 ?
0
4 -> 1 ?
1
4 -> 2 ?
1
4 -> 3 ?
1
Graph:
|0|1|0|1|0|
|1|0|1|0|1|
|0|1|0|1|1|
|1|0|1|0|1|
|0|1|1|1|0|
Colors count:

```

(۴) تعیین تعداد رنگ ها. پس از نمایش ماتریس تعداد رنگ ها از کاربر سوال می شود که در شکل بالا ملاحظه می کنید.

(۵) انجام فرایند رنگ آمیزی. پس از دریافت تمام اطلاعات پیش نیاز برنامه شروع به پردازش می کند و از طریق روش **backtracking** رئوس گراف را رنگ آمیزی می کند. همانطور که در شکل زیر ملاحظه می کنید در طی این فرایند همه رئوس تک به تک بررسی شده و در طی فرایند هر جا **backtracking** اتفاق افتد نمایش داده می شود.

```

Operation starts...
Checking vertex 0
  Checking color 1 for Vertex 0
  Color 1 assigned for Vertex 0

Checking vertex 1
  Checking color 1 for Vertex 1
  BackTrack...
  Checking color 2 for Vertex 1
  Color 2 assigned for Vertex 1

Checking vertex 2
  Checking color 1 for Vertex 2
  Color 1 assigned for Vertex 2

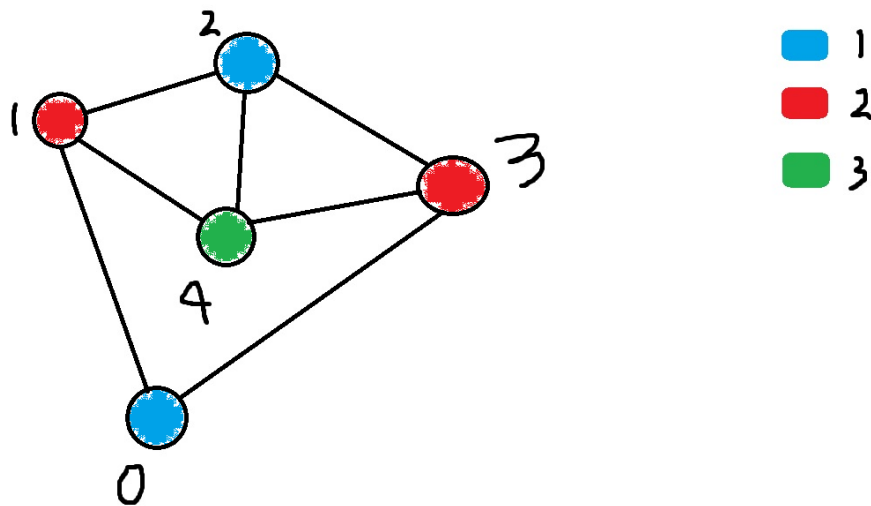
Checking vertex 3
  Checking color 1 for Vertex 3
  BackTrack...
  Checking color 2 for Vertex 3
  Color 2 assigned for Vertex 3

Checking vertex 4
  Checking color 1 for Vertex 4
  BackTrack...
  Checking color 2 for Vertex 4
  BackTrack...
  Checking color 3 for Vertex 4
  Color 3 assigned for Vertex 4

Solution:
Vertex 0 -> Color: 1
Vertex 1 -> Color: 2
Vertex 2 -> Color: 1
Vertex 3 -> Color: 2
Vertex 4 -> Color: 3

```

۶) اعلام راه حل. در آخر پس از پایان عملیات در صورت وجود جواب برای مسئله به ازای هر رأس یک رنگ اعلام می گردد. راه حل برنامه برای گراف مورد نظر به صورت زیر است.



کد برنامه:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

namespace ABTProject
{
    public class GraphColoring
    {

```

```

int _vertexNo;

public GraphColoring(int vertexNo)
{
    _vertexNo = vertexNo;
}

int[] color;

public bool CheckLoacalView(int v, int[][] graph, int[] color, int c)
{
    for (int i = 0; i < _vertexNo; i++)
    {
        if (graph[v][i] == 1 && c == color[i])
        {
            Console.WriteLine("    BackTrack...");
            return false;
        }
    }

    return true;
}

public bool HandleOk(int[][] graph, int m, int[] color, int v)
{
    if (v == _vertexNo)
    {
        return true;
    }

    Console.WriteLine("Checking vertex " + v);

    for (int c = 1; c <= m; c++)
    {
        Console.WriteLine("    Checking color " + c + " for Vertex " + v);
        if (CheckLoacalView(v, graph, color, c))
        {
            color[v] = c;
            Console.WriteLine("        Color " + c + " assigned for Vertex " + v
+ Environment.NewLine);

            if (HandleOk(graph, m, color, v + 1))
            {
                return true;
            }

            color[v] = 0;
        }
    }

    return false;
}

public bool DoColoring(int[][] graph, int m)
{
    color = new int[_vertexNo];

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

```

```

    {
        color[i] = 0;
    }

    if (!HandleOk(graph, m, color, 0))
    {
        Console.WriteLine("Solution does not exist");
        return false;
    }

    PrintSolution(color);
    return true;
}

public void PrintSolution(int[] color)
{
    Console.WriteLine("Solution:");

    for (int i = 0; i < _vertexNo; i++)
    {
        Console.WriteLine("    Vertex " + i + " -> Color: " + color[i] + " ");
    }

    Console.WriteLine();
}
}
}

```

برنامه را می توانید از مسیر ABTProject/ bin/Debug/ABTProject.exe اجرا نمایید.