

Practical no: 8

Name: Saloni Vishwakarma

Batch-Roll no: C1-13

Subject: DAA Lab

Date of execution: 7 July 2023

Aim: To implement Graph Coloring algorithm using recursion in C.

Problem statement: A GSM is a cellular network (shown in fig) with its entire geographical range divided into hexadecimal cells. Each cell has a communication tower which connects with mobile phones within cell. Assume this GSM network operates in only four (1-4) different frequency ranges. Allot frequencies to each cell such that no adjacent cells have same frequency range. (use Graph coloring concept)

Graph Coloring Algorithm (Code and Output):

```
#include <stdio.h>
#define MAX 100
int graph[MAX][MAX];
int colors[MAX];
int ver, numColors;

// Function to check if it is safe to color a vertex with a given color
int isSafe(int v, int color) {
    int i;
    for (i = 0; i < ver; i++) {
        if (graph[v][i] && colors[i] == color)
            return 0;
    }
    return 1;
}

// Function to recursively apply backtracking to color the graph
int graphColoringUtil(int v) {
    if (v == ver)
        return 1;
}
```

```

int color;
for (color = 1; color <= numColors; color++) {
    if (isSafe(v, color)) {
        colors[v] = color;

        if (graphColoringUtil(v + 1))
            return 1;

        colors[v] = 0; // Backtracking
    }
}
return 0;
}

```

```

// Function to print the colored graph
void printColoredGraph() {
    printf("\n Vertex\tColor\n");
    int i;
    for (i = 0; i < ver; i++) {
        printf(" %d \t %c\n", i + 1, '@'+colors[i]);
    }
}

```

```

int main() {
    printf("\n Enter the number of vertices: ");
    scanf("%d", &ver);

    printf("\n Enter the adjacency matrix:\n");
    int i, j;
    for (i = 0; i < ver; i++) {
        printf(" ");
        for (j = 0; j < ver; j++) {
            scanf("%d", &graph[i][j]);
        }
    }
    printf("\n Enter the number of colors: ");
    scanf("%d", &numColors);

    if (graphColoringUtil(0)) {

```

```
    printf("\n Graph coloring is possible with %d colors.\n", numColors);  
    printColoredGraph();  
} else {  
    printf("\n Graph coloring is not possible with %d colors.\n", numColors);  
}  
return 0;  
}
```

```
Enter the number of vertices: 5
```

```
Enter the adjacency matrix:
```

```
0 1 1 0 1  
1 0 1 0 1  
1 1 0 1 0  
0 0 1 0 1  
1 1 0 1 0
```

```
Enter the number of colors: 4
```

```
Graph coloring is possible with 4 colors.
```

```
Vertex Color
```

```
1      A  
2      B  
3      C  
4      A  
5      C
```

```
...Program finished with exit code 0  
Press ENTER to exit console.
```

```
Enter the number of vertices: 5

Enter the adjacency matrix:
0 1 1 0 1
1 0 1 0 1
1 1 0 1 0
0 0 1 0 1
1 1 0 1 0

Enter the number of colors: 2

Graph coloring is not possible with 2 colors.

...Program finished with exit code 0
Press ENTER to exit console.
```

Conclusion: We have successfully studied and implemented Graph Coloring Algorithm using backtracking(recursion) in C.