Name- Ratika

University Roll No - 1121109

Class Roll No - 28

Course - BCA Sem - 6th

Section- B

Paper Name- Computer Graphics and animation

Type of paper- Regular Endterm Practical

P1- Write an algorithm and program to implement Floodfill Algorithm using 8 connected Approach.

```
#include <stdio.h>
#include <graphics.h>
#include <dos.h>
#include <conio.h>
Void floodfill (int x, inty, intold, intnewcol)
{
    int current;
    current = getpixel (x,y);
    if (current == old)
    {
        delay(5);
        putpixel (x, y, newcol);
        floodfill (x+1, y, old, newcol);
        floodfill(x-1, y, old, newcol);
        floodfill (x, y+1, old, newcol);
        floodfill (x, y-1, old newcol);
        floodfill (x+1, y+1, old, newcol);
```

Rutika

```
        floodfill(x-1,y+1,old,newcol);
        floodfill(x+1,y-1,old,newcol);
        floodfill(x-1,y-1,old,newcol);

    }

}
void main()
{
    int gd = DETECT, gm;
    initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
    rectangle(50,50,150,150);
    floodfill(70,70,0,15);
    getch();
    closegraph();

}
```

Rutika

Algorithm:-

Step 1:- Start

Step 2:- Initialize the value of seed point (x, y, old, newcol).

Step 3:- Define the boundary values.

Step 4:- Check if the current seed point is of default color then repeat the steps 4 and 5 till the boundary pixels reached.

```
if (current == old)
```

Step 5:- Recursively following the below procedure:-

```
Procedure floodfill(x, y : fill_color,
                    old_color : integer)
if(getpixel(x, y) = old_color)
{
```

Ratika

```
setpixel(x, y, fill_color);
fill(x+1, y, fill_color, old_color);
fill(x-1, y, fill_color, old_color);
fill(x, y+1, fill_color, old_color);
fill(x, y-1, fill_color, old_color);
}
}
```

Step 6: STOP

Ratika