Program of Boundary fill algorithm:-

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<doc.h>

void boundary\_fill(int x, int y, int fcolor, int bcolor)

{

if ((getpixel(x, y) != bcolor) && (getpixel(x, y) != fcolor))

{

delay(10);

putpixel(x, y, fcolor);

boundary\_fill(x + 1, y, fcolor, bcolor); boundary\_fill(x , y+1, fcolor, bcolor); boundary\_fill(x+1, y + 1, fcolor, bcolor); boundary\_fill(x-1, y - 1, fcolor, bcolor); boundary\_fill(x-1, y, fcolor, bcolor); boundary\_fill(x , y-1, fcolor, bcolor); boundary\_fill(x-1, y + 1, fcolor, bcolor); boundary\_fill(x+1, y - 1, fcolor, bcolor);

}

}

void main()

{

int x, y, fcolor, bcolor;

int gd=DETECT,gm;

initgraph(&gd, &gm, "C:\\TurboC3\\BGI"); printf("Enter the seed point (x,y) : "); scanf("%d%d", &x, &y);

printf("Enter boundary color : "); scanf("%d", &bcolor); printf("Enter new color : ");

scanf("%d", &fcolor); rectangle(50,50,100,100); boundary\_fill(x,y,fcolor,bcolor); getch();

}

Output:-



Program for flood fill algorithm:-

#include<stdio.h>

#include<graphics.h> #include<dos.h>

void flood(int,int,int,int); int main()

{

int gd,gm=DETECT; //detectgraph(&gd,&gm); initgraph(&gd,&gm," "); rectangle(50,50,100,100); flood(55,55,12,0); closegraph(); return 0;

} void flood(int x,int y, int fill\_col, int old\_col)

{

if(getpixel(x,y)==old\_col)

{ delay(10); putpixel(x,y,fill\_col); flood(x+1,y,fill\_col,old\_col); flood(x-1,y,fill\_col,old\_col); flood(x,y+1,fill\_col,old\_col); flood(x,y-1,fill\_col,old\_col); flood(x + 1, y + 1, fill\_col, old\_col); flood(x - 1, y - 1, fill\_col, old\_col); flood(x + 1, y - 1, fill\_col, old\_col); flood(x - 1, y + 1, fill\_col, old\_col);

}

}

Output:-

