Ans 1.

Name - Pranav Joshi
Course - BCA Sec - B
Roll no - 1121104 (23)
Subject - Computer Graphics

Algorithm

Step 1 - Start

Step 2 - Initialize the graphics mode.

Step 3 - ~~construct~~ Draw the rectangle using rectangle function.

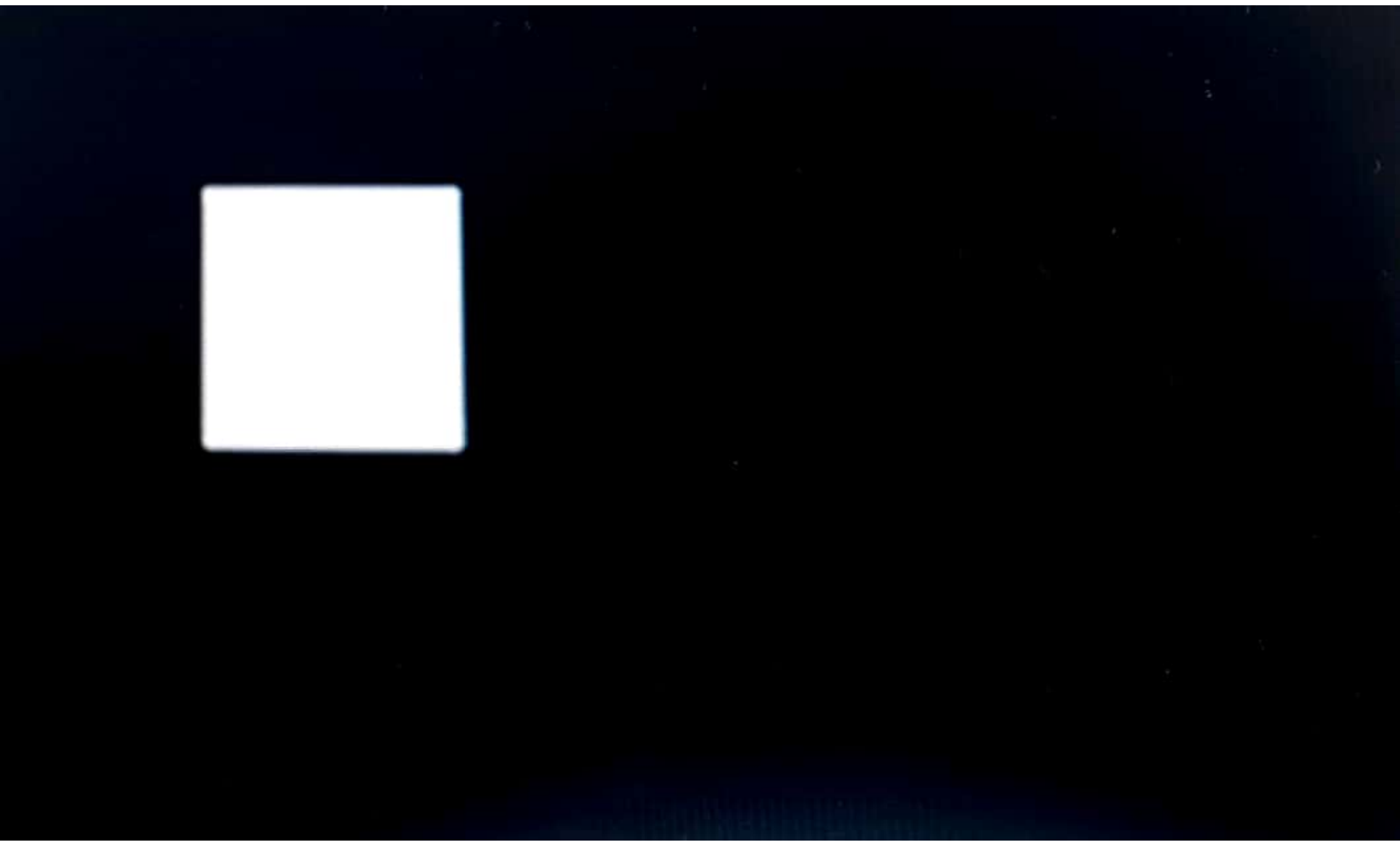
step 4 - Implement 8 connected floodfill with the co-ordinates x and y.

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

Step 5 - Stop

Program

```
#include <stdio.h>
# include <graphics.h>
# include <conio.h>
void floodfill (int x, int y, int old, int newcol)
{
  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);
    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);
  rectangle (50, 50, 150, 150);
  floodfill (70, 70, 0, 15);
  getch();
  closegraph();
}
```



Name- Pranav Joshi
Course- BCA Sec.- B
Roll no.- 1121104(23)
sub.- Computer Graphics

Ans 3.

Algorithm

Step 1 - Start

Step 2 - Declare p, q, r, x, y, d variables
p, q are coordinates of the center of circle
r is the radius of circle.

Step 3 - Enter value of r.

Step 4. calculate $d = 3 - 2r$

Step 5 - Initialize $x = 0$ $y = r$

Step 6 - check if the whole circle is scan converted
If $x >= y$
stop.

step 7 - Plot eight points by using concepts of eight-way symmetry.
The center is (p, q). Current active is (x, y)

putpixel (x+p, y+q)
putpixel (y+p, x+q)
putpixel (-y+p, x+q)
putpixel (-x+p, y+q)
putpixel (-x+p, -y+q)
putpixel (-y+p, -x+q)
putpixel (y+p, -x+q)
putpixel (x+p, -y-q)

step 8 - Find location of next pixels to be scanned.
If $d < 0$
then $d = d + 4x + 6$
increment $x = x + 1$
If $d \geq 0$
then $d = d + 4(x - y) + 10$

increment $x = x+1$

decrement $y = y+1$

step9- Go to step 6

step10- stop.

Program

```
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
#include <math.h>
void EightWaysymmetricPlot (int xc, int yc, int x, int y)
{
putpixel (x+xc, y+yc, RED);
putpixel (x+xc, -y+yc, YELLOW);
putpixel (-x+xc, -y+yc, GREEN);
putpixel (-x+xc, y+yc, YELLOW);
putpixel (y+xc, x+yc, 12);
putpixel (y+xc, -x+yc, 14);
putpixel (-y+xc, -x+yc, 15);
putpixel (-y+xc, x+yc, 6);
}
void BresenhamCircle (int xc, int yc, int r)
{
int x=0, y=r, d=3-(2*r);
EightWaysymmetricPlot (xc, yc, x, y);
while (x<=y)
{
if (d<=0)
{
```

```
        d=d+(4*x)+6;
    }
  else
{
d=d+(4*x)-(4*y)+10;
   y=y+1;
}
 x=x+1;
FightWaySymmetricPlot(xc,yc,x,y);
}
}
 int main(void)
{
int xc,yc,r,gdriver=DETECT,gmode,errorcode;
initgraph(&gdriver,&gmode);
 errorcode=graphresult(0);
 if (errorcode!=grok)
{
printf("Graphics error:%s\n",grapherrormsg(errorcode));
printf("Press any key halt:");
getch();
exit(1);
}
printf("Enter values of xc And yc:");
scanf("%d %d",&xc,&yc);
printf("Enter the value of radius:");
scanf("%d",&r);
getch();
closegraph();
return 0;
}
```