Name - Naveen Singh
Rollno - 1121087
Section - B

1. Algorithm

floodfill (x, y, oldcolor, newcolor)

1) If x or y is outside the screen, then return n.

2. If color of getpixel(x,y) is same as oldcolor, then for loop

3. Recur for

```
floodfill (x, y, oldcolor, newcolor)
floodfill (x+1, y, oldcolor, newcolor)
floodfill (x-1, y, oldcolor, newcolor)
floodfill (x, y-1, oldcolor, newcolor)
floodfill (x+1, y+1, oldcolor, newcolor)
floodfill (x-1, y+1, oldcolor, newcolor)
floodfill (x+1, y-1, oldcolor, newcolor)
floodfill (x-1, y-1, oldcolor, newcolor)
```

Scanned with CamScanner

Name- Naveen Singh

1.

```
# 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, "C:\\TURBOC3\\BGI"))
   rectangle(50, 50, 150, 150);
   floodfill (70, 70, 0, 15);
   getch ();
   closegraph();
 }
```

[redacted]

Name - Naveen Singh
Section - B

3.

```
# include <stdio.h>
# include < graphics.h>

void drawcircle (int xc, int yc, int x, int y)
{
    putpixel (xc+x, yc+y, Red);
   putpixel (xc-x, yc+y, Red);
   putpixel (xc+x, yc-y, Red);
   putpixel (xc-x, yc-y, Red);
   putpixel (xc+y, yc+x~~3~~, Red);
   putpixel (xc~~+~~-y, yc+x, Red);
   putpixel (xc+y, yc-x3, Red);
   putpixel (xc-y, yc-x, Red);
 3

void CircleBres (int xc, int yc, int r)
  {
   int x= 0, y= r;
   int d= 3-2 * r;
```

```
drawcircle (xc, yc, x, y);
while (y>=x)
{
  x++;
  if (d>0)
 {
 y--;
 d=d+4*(x-y)+10;
 }
 else
  d=d+4*x+6;
  drawcircle (xc, yc, x, y);
  delay (50);
  }
  }
 int main ()
 {
 int XC=50, yc=50, r2=30;
 int gd= Detect, gm;
 initgraph (&gd, &gm, "");
 circleBres(xc, yc, r);
return 0;
}
```

Name- Naveen Singh

Section - B

Rollno. - 1121087

Algorithm

step 1: set initial values of (xc, yc) and (x, y)

step 2: set decision parameter d to $d = 3-(2*r)$.

step 3: call draw circle (int xc, int yc, int x, int y) function.

step 4: Repeat step 5 to 8 until $x <= y$.

step 5: Increment value of x.

step 6: if $d < 0$, set $d = d + (4*x) + 6$

step 7: Else, set $d = d + 4*(4x-y) + 10$ and decrement y by 1.

step 8: Call drawcircle (int xc, int yc, int x, int y) function.

step 9: Exit