NAME:- Vinay Singh Negi

Class Roll 51

UNIVERSITY Roll No.:- 1121166

PAPER CODE:- PBC 602

PAPER NAME:- Computer graphics

Section:- C

Q1)

Ans 1.) Bresenham's Line Algorithm:

Step1:- Start Algorithm

Step 2:- Declare x_1, x_2, Y_1, y_2, d, i_1, i_2, dx, dy

Step3 :- Enter value of x_1, y_1, x_2, Y_2

where x_1, Y_1 are coordinates of starting point

And X_2, Y_2 are coordinates of Ending point

Step4 :- Calculate $dx = x_2 - x_1$

Calulate $dy = Y_2 - y_1$

calculate $i_1 = 2 * dy$

calculate $i_2 = 2 * (dy - dx)$

Calculate $d = i_1 - dx$

Vinay

```
Step 5: consider $(x, y)$ as Starting point
  and $x_{end}$ as maximum possible value of x
    if $dx < 0$
        Then $x = x_2$
        $y = y_2$
        $x_{end} = x_1$
    if $dx > 0$
        Then $x = x_1$
        $y = y_1$
        $x_{end} = x_2$
Step 6: Generate point at $(x, y)$ coordinates
step 7:- check if whole line is generated.
    if $x >= x_{end}$
        Stop.
Step 8:- calculate co-ordinate of next pixel
    If $d < 0$
        Then $d = d + i_1$
    if $d \geq 0$
        Then $d = d + i_2$
        increment $y = y + 1$
```

Vinay

Step 9: Increment x = x+1

Step 10: Draw a point of latest (x,y) coordinates

Step 11: Go to step 7

Step 12: End of Algorithm

code

```
 #include<stdio.h>
 #include<graphics.h>
 Void drawline (int x0, int y0, int x1 , int y1)
 {
  int dx, dy,P,x,y;
  dx=x1-x0;
  dy=y1-y0;
  x=x0;
  y=y0;
  P=2*dy-dx
  while(x<x1)
  {
  if(P>=0)
   {
    Putpixel(x,y,7);
```

Vinay

```
y = y + 1 ;
p = p + 2 * dy - 2 * dx ;

}
else
  {
  putpixel (x, y, 7);
   p = p + 2 * dy;
   }
  x = x + 1 ;

   }
 }
int main ()
{ int gdriver = DETECT, gmode, error, x0, y0, x1, y1;
initgraph (&gdriver, &gmode, " ");
Printf ("Enter co-ordinates of first point:");
Scanf ("%d %d", &x0, &y0);
Printf ("Enter co-ordinates of second point:");
scanf ("%d %d", &x1, &y1);
drawline (x0, y0, x1, y1);
return 0;
 }
```

Binary

Q2.)

```
Step 1 : Put x=0, y=r in eqn 2
          we have P=1-r

Step 2 :  Repeat Steps while x ≤ y
          Plot (x,y)
           if (P<0)
        Then set P = P + 2x + 3
        Else
        P = P+ 2(x-y)+5
         y = y-1 (endif)
          x = x+1 (end loop)
  Step3 : End
```

#code

```
  #include <stdio.h>
  #include <graphics.h>
 void drawcircle (int x0, int y0, int radius)
  {
   int x= radius;
    int y=0;
    int err = 0;
    while (x>=y)
    {
```

```
{ putpixel(x0+x, y0+y,7);
  putpixel (x0+y, y0+x,7);
  putpixel ( x0-y, y0+x,7));
  putpixel (x0-x, y0+y, 7))
  putpixel (x0-x, y0-y, 7));
  putpixel (x0,-y, y0-x, 7));
  putpixel (x0+y, y0-x, 7));
  putpixel ( x0+x, y0-y, 7)),

  if (err <=0)
  {
   y +=1;
   err +=2*y+1;
  }
  if(err>0)
  {
   x -=1
   err -= 2*x+1;
  }
 }
}
int main()
{
int gdriver = DETECT, gmode, error, x, y, r;
printf ("Enter radius ofcircle:");
scanf ("%d", &r);
```

```
Printf (" Enter co-ordinates of center
                              (x and y):");
 Scanf ("%d %d", &x, &y);
 initgraph (&gdriver, &gmode, "");
 drawcircle (x, y, r);
 delay (99999);
   return 0;
  }
```