NAME ⇒ VISUAL UJJWAL

COURSE ⇒ BCA 'C' '57'

U. ROLL NO ⇒ 1121172

SUBJECT ⇒ CG PRACTICAL TEST

Ans1

```
# include <stdio.h>
# include <graphic.h>
void draw line (int xo, int yo, 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)
   {  put pixel (x,y,7);
        y = y+1;
      P = P+2* dy-2*dx;
}
     else
   {   put pixel (x,y,7);
       P = P+2*dy;
      }
       x = x+1;
     }
  }
      int main()
  {
```

```
int gdrive= DETECT, gmode ,error, x0, y0, x1, y1;
  initgraph (&gdriver, &gmode, "C\\ turboC3\\bgi

   Printf ("Enter Coordinates 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).

        return0;
   }
```

ALGORITUM ÷

STEP 1 ÷

- This Algorithm is used in Computer graphic for drawing Line for Scan Converting a line

STPEP ÷ 2

- Assome a pixel P_1 (x_1, y_1), then Select subsequent pixels

STEP ÷ 3

- Either the One to its right (lower-bound for the line

STEP 4 ÷

- One top its right and up (upper bound for the line]

STEP 5 ÷

if S we have $x_{i+1} = x_i + 1$, $y_{i+1} = y_1$

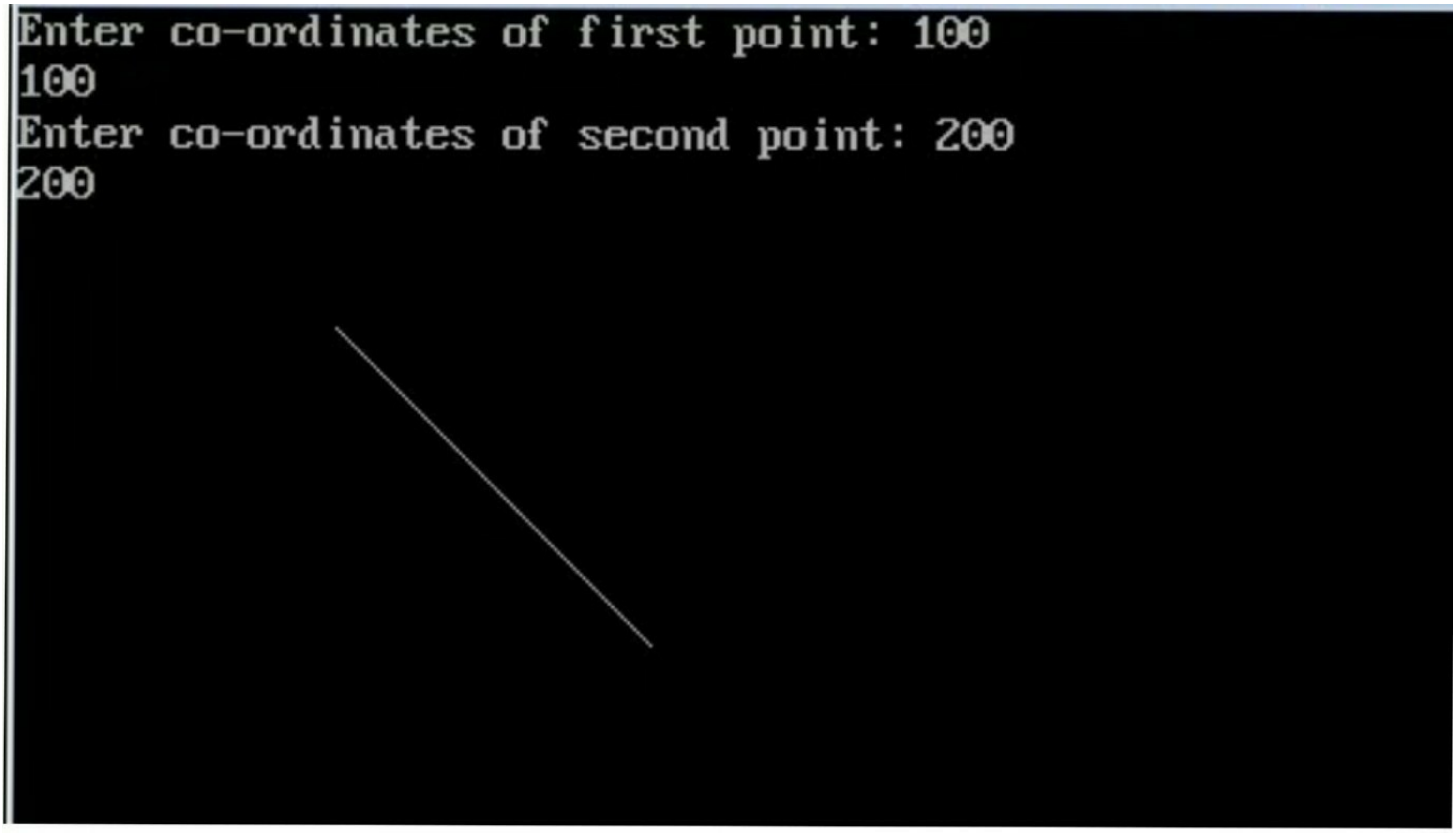
if T is chosen we have $x_{i+1} = x_i + 1$ and $y_{i+1} = y_i + 1$

STEP 6 ÷ when $(s-t) < 0 \Rightarrow S < t$ → The closet pixel is S

when $(s-t) \geq 0 \Rightarrow S < t$

The Closet pixel is T

STEP 7 ÷ End



```
Enter co-ordinates of first point: 100
100
Enter co-ordinates of second point: 200
200
```

Ans2

Algorithm to mid-point Circle drawing Algori

```
Step 1 ÷  put $x = 0$, $y = r$ in eq ①

        $P_i = f(x_{i+1}, y_i - \frac{1}{2}) = (x_i + 1)^2 + (y_i - \frac{1}{2})^2 - r^2 \to ①$

      We ~~which~~ have   $P = 1 - r$

Step 2 ÷   Repeat steps while $x \leq y$

           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)

Step 3 ÷   End
```

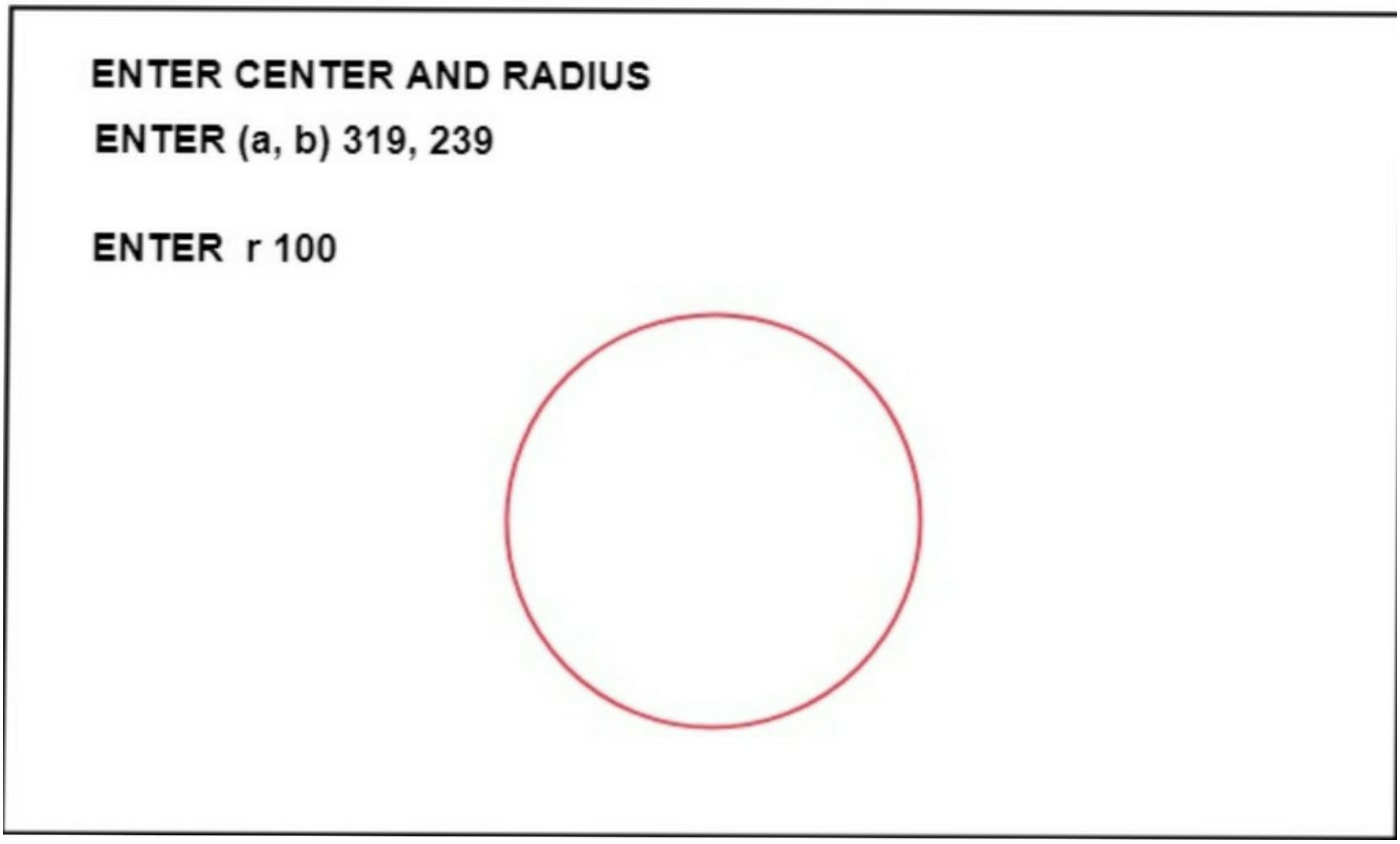
Ans2 Program to implenent mid-point Circle using midpoint Algorithm

```
#include < graphics.h>
# include <stdlib.h>
# include < math.h >
# include < conio.h>
# include <iostream.h >

Class bresen
{
    float x, y, a, b, r, p;
    Public:
    void get ();
    void Cal ();
}
    Void main ()
{
    bresen b;
b.get ();
b.cal ();
getch ();
}
    void bresen :: get ()
{
    Cout << "Enter Center and Radius";
    Cout << "Enter (a,b)";
     Cin >> a >> b
    Cout << "Enter r";
     Cin >> r;
 }
   void bresen :: cal ()
```

```
    {
     int gdriver = DETECT, gmode, errorcode;
        int midx, midy , i;
    initgraph (&gdriver , &gmode,"];
if  (errorcode != grok)
  {
     Print f ("Graphic error: %s \n", grapherrorm
          (errorcode);

   Print f (" press any key to halt: ");
         getch ();
      exit (1)
  }
      x = 0;
      y = r;
 Put pixel (a, b+r, RED).
 Put pixel (a, b-r, RED);
 Put pixel (a-r, b, RED);
 Put pixel (a+r, b, RED);
   P = 5/4) - r.
While (x<=y)
{   if (P<0)
    P+ = (4* x)+6;
  else
     {
    P+= (2*(x-y)+5;
       y--;
   }  x++;
  Put pixel (a+x , b+y, RED);
 Put Pixel (a-x , b+y , RED).
 Put  Pixel ( a+x , b-y , RED);
 Put pixel (a+x , b-y , RED);
 Put pixel (a-x , b+y , RED);
 Put pixel ( a-x, b-y, RED);
}
}
```



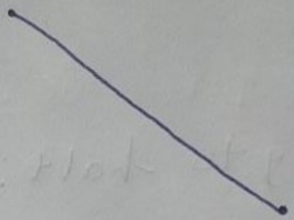
```
ENTER CENTER AND RADIUS
ENTER (a, b) 319, 239

ENTER  r 100
```

Out put of Breshnam line Drawing Algorithm :

Enter Co-ordinates of First point : 100
100

Enter Coordinates of Second Point : 200
200



Output of mid point Circle Drawing Algorithm:

Enter Center and Radius

Enter (a,b) 319,239

Enter r100