5. Solve the following

a. Develop the program for the mid-point circle drawing algorithm.

b. Develop the program for the mid-point ellipse drawing algorithm.

a. Develop the program for the mid-point circle drawing algorithm.

Solution:-

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
void main()
{
int x,y,x_mid,y_mid,radius,dp;
int g_mode,g_driver=DETECT;
clrscr();
initgraph(&g_driver,&g_mode,"C:\\TURBOC3\\BGI");
printf("********** MID POINT Circle drawing algorithm ********\n\n");
printf("\nenter the coordinates= ");
scanf("%d %d",&x_mid,&y_mid);
printf("\n now enter the radius =");
```

```
scanf("%d",&radius);
x=0;
y=radius;
dp=1-radius;
do
{
putpixel(x_mid+x,y_mid+y,YELLOW);
putpixel(x_mid+y,y_mid+x,YELLOW);
putpixel(x_mid-y,y_mid+x,YELLOW);
putpixel(x_mid-x,y_mid+y,YELLOW);
putpixel(x_mid-x,y_mid-y,YELLOW);
putpixel(x_mid-y,y_mid-x,YELLOW);
putpixel(x_mid+y,y_mid-x,YELLOW);
putpixel(x_mid+x,y_mid-y,YELLOW);
if(dp<0) {
dp+=(2*x)+1;
}
else{
y=y-1;
dp+=(2*x)-(2*y)+1;
}
x=x+1;
```

```
}while(y>x);
getch();
}
```

b. Develop the program for the mid-point ellipse drawing algorithm.

Solution:-

```
#include<conio.h>
#include<dos.h>
#include<stdio.h>
#include<graphics.h>
void main(){
     long x,y,x_center,y_center;
     long a_sqr,b_sqr, fx,fy, d,a,b,tmp1,tmp2;
     int g_driver=DETECT,g_mode;
     clrscr();

   initgraph(&g_driver,&g_mode,"C:\\TURBOC3\\BGI");
   printf("********* MID POINT ELLIPSE ALGORITHM
*********");
   printf("\n\n Enter coordinate x and y = ");
```

```
 scanf("%ld%ld",&x_center,&y_center);
 printf("\n Now enter constants a and b = ");
 scanf("%ld%ld",&a,&b);
 x=0;
 y=b;
 a_sqr=a*a;
 b_sqr=b*b;
 fx=2*b_sqr*x;
 fy=2*a_sqr*y;
d=b_sqr-(a_sqr*b)+(a_sqr*0.25);
do
 {
putpixel(x_center+x,y_center+y,1);
putpixel(x_center-x,y_center-y,1);
putpixel(x_center+x,y_center-y,1);
putpixel(x_center-x,y_center+y,1);

 if(d<0)
  {
d=d+fx+b_sqr;
  }
 else
```

```
{
y=y-1;
d=d+fx+-fy+b_sqr;
fy=fy-(2*a_sqr);
}
x=x+1;
fx=fx+(2*b_sqr);
delay(10);

 }
 while(fx<fy);
 tmp1=(x+0.5)*(x+0.5);
 tmp2=(y-1)*(y-1);
 d=b_sqr*tmp1+a_sqr*tmp2-(a_sqr*b_sqr);
 do
 {
putpixel(x_center+x,y_center+y,1);
putpixel(x_center-x,y_center-y,1);
putpixel(x_center+x,y_center-y,1);
putpixel(x_center-x,y_center+y,1);

 if(d>=0)
```

```
  d=d-fy+a_sqr;
   else

  {
  x=x+1;
  d=d+fx-fy+a_sqr;
  fx=fx+(2*b_sqr);
  }
   y=y-1;
   fy=fy-(2*a_sqr);
   }
   while(y>0);
   getch();
   closegraph();
}
```