Implement midpoint Ellipse Algorithm

```
#include<stdio.h>

#include<graphics.h>

#include<conio.h>

#include<dos.h>


int 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*");

        printf("\n Enter coordinate x = ");

        scanf("%ld",&x_center);

        printf(" Enter coordinate y = ");

        scanf("%ld",&y_center);

        printf("\n Now Enter constants a =");

        scanf("%ld",&a,&b);

        printf(" Now Enter constants b =");

        scanf("%ld",&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,5);
        putpixel(x_center-x,y_center+y,5);

        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(1000);
}
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,4);
        putpixel(x_center-x,y_center-y,4);
        putpixel(x_center+x,y_center-y,2);
        putpixel(x_center-x,y_center+y,2);

        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();
        return 0;
}
```