// Experiment No. 2: Midpoint Ellipse Drawing Algoritm

```
#include <conio.h>
#include <stdio.h>
#include <graphics.h>
void main()
{
    int gd = DETECT, gm;
    float x, y, xc, yc, rx, ry, pk, pk1;
    clrscr();

    initgraph(&gd, &gm, "C:\\TURBOC3\\BGI"); //Paste your BGI path here
    cleardevice();

    printf("Mid Point Ellipse Drawing Algorithm\n");

    printf("Enter Center for ellipse: \nx : ");
    scanf("%f", &xc);

    printf("y : ");
    scanf("%f", &yc);

    printf("Enter x-radius and y-radius: \nx-radius : ");
    scanf("%f", &rx);

    printf("y-radius : ");
    scanf("%f", &ry);

    x = 0;
    y = ry;

    pk = (ry * ry) - (rx * rx * ry) + ((rx * rx) / 4);

    while ((2 * x * ry * ry) < (2 * y * rx * rx))
    {
        if (pk <= 0)
        {
            x = x + 1;
            pk1 = pk + (2 * ry * ry * x) + (ry * ry);
        }
        else
        {
            x = x + 1;
            y = y - 1;
            pk1 = pk + (2 * ry * ry * x) - (2 * rx * rx * y) + (ry * ry);
        }
```

```
        pk = pk1;

        setbkcolor(WHITE);
        putpixel(xc + x, yc + y, 2);
        putpixel(xc - x, yc + y, 2);
        putpixel(xc + x, yc - y, 2);
        putpixel(xc - x, yc - y, 2);
    }

    pk = ((x + 0.5) * (x + 0.5) * ry * ry) + ((y - 1) * (y - 1) * rx * rx) - (rx * rx * ry * ry);

    while (y > 0)
    {
        if (pk > 0)
        {
            y = y - 1;
            pk1 = pk - (2 * rx * rx * y) + (rx * rx);
        }
        else
        {
            x = x + 1;
            y = y - 1;
            pk1 = pk + (2 * ry * ry * x) - (2 * rx * rx * y) + (rx * rx);
        }

        pk = pk1;

        setbkcolor(WHITE);
        putpixel(xc + x, yc + y, 2);
        putpixel(xc - x, yc + y, 2);
        putpixel(xc + x, yc - y, 2);
        putpixel(xc - x, yc - y, 2);
    }
    setbkcolor(WHITE);
    setcolor(RED);
    line(xc + rx, yc, xc - rx, yc);
    line(xc, yc + ry, xc, yc - ry);

    setcolor(RED);
    outtextxy(xc + (1.2 * rx), yc - (1.2 * ry), "(x,y)");
    outtextxy(xc - (1.2 * rx), yc + (1.2 * ry), "(-x,-y)");
    outtextxy(xc + (1.2 * rx), yc + (1.2 * ry), "(x,-y)");
    outtextxy(xc - (1.2 * rx), yc - (1.2 * ry), "(-x,y)");
    getch();
}
```

OUTPUT:

Mid Point Ellipse Drawing Algorithm

Enter Center for ellipse:
x: 200
y: 200

Enter x-radius and y-radius:
x-radius : 60
y-radius : 60