****

**Experiment Name -1:** **Using midpoint algorithm draw circle.**

**Code:**

#include<graphics.h>

#include<bits/stdc++.h>

using namespace std;

int main()

{

float x,y,xc,yc,r;

int gd=DETECT;

int gm;

initgraph(&gd,&gm,"");

cout<<"Enter center of circle : ";

cin>>xc>>yc;

cout<< "Enter radius of circle : ";

cin>>r;

x=0;

y=r;

int d=1-r;

do {

putpixel(xc + x, yc + y, WHITE);

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

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

putpixel(xc - x, yc - y, WHITE);

putpixel(xc + y, yc + x, WHITE);

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

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

putpixel(xc - y, yc - x, WHITE);

if (d < 0)

d += 2 \* x + 3;

else {

d += 2 \* (x - y) + 5;

y--;

}

x++;

} while (x <= y);

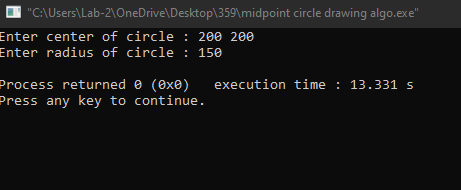
delay(5000);

getch();

closegraph();

}

**Output:**





**Experiment Name -2:** **Ellipse drawing using midpoint algorithm.**

**Code:**

#include <bits/stdc++.h>

#include <graphics.h>

#include <math.h>

using namespace std;

void Ellipse(int xc, int yc, int a, int b)

{

int x = 0;

int y = b;

int a\_Sqrt = a \* a;

int b\_Sqrt = b \* b;

int twoaSqrt = 2 \* a\_Sqrt;

int twobSqrt = 2 \* b\_Sqrt;

int xEnd = round(a\_Sqrt / sqrt(a\_Sqrt + b\_Sqrt));

int p = round(b\_Sqrt - a\_Sqrt \* b + 0.25 \* a\_Sqrt);

while (x <= xEnd)

{

putpixel(xc + x, yc + y, WHITE);

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

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

putpixel(xc - x, yc - y, WHITE);

x++;

if (p < 0)

{

p += twobSqrt \* x + b\_Sqrt;

}

else

{

y--;

p += twobSqrt \* x - twoaSqrt \* y + b\_Sqrt;

}

}

p = round(b\_Sqrt \* (x + 0.5) \* (x + 0.5) + a\_Sqrt \* (y - 1) \* (y - 1) - a\_Sqrt \* b\_Sqrt);

while (y >= 0)

{

putpixel(xc + x, yc + y, WHITE);

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

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

putpixel(xc - x, yc - y, WHITE);

y--;

if (p > 0)

{

p += a\_Sqrt - twoaSqrt \* y;

}

else

{

x++;

p += twobSqrt \* x - twoaSqrt \* y + a\_Sqrt;

}

}

}

int main()

{

int gd = DETECT, gm;

initgraph(&gd, &gm, "");

int xc = 250; // X-coordinate of the center

int yc = 250; // Y-coordinate of the center

int a = 100; // Semi-major axis length

int b = 50; // Semi-minor axis length

printf("Enter X-coordinate of the center: ");

cin>>xc;

printf("Enter Y-coordinate of the center: ");

cin>>yc;

printf("Enter Semi-major axis length: ");

cin>>a;

printf("Enter Semi-minor axis length: ");

cin>>b;

Ellipse(xc, yc, a, b);

delay(50000);

closegraph();

return 0;

}

**Output:**

