*Course title: Computer Graphics Lab*

*Course code: CSE-304*

*3rd Year 1st Semester Examination 2022*

**Date of Submission**: 4 June 2023

****

**Submitted to-**

**Dr. Mohammad Shorif Uddin**

*Professor*

**Dr. Morium Akter**

*Associate Professor*

*Department of Computer Science and Engineering*

*Jahangirnagar University*

*Savar, Dhaka-1342*

| **Sl** | Class Roll | Exam Roll | Name |
| --- | --- | --- | --- |
| 01 | 382 | 202194 | Mohammed Tamjid Islam |

**Mid Point Circle:**

#include <iostream>

#include <graphics.h>

void drawCircle(int xc, int yc, int radius)

{

int x = 0;

int y = radius;

int p = 1 - radius;

while (x <= y)

{

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

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

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

if (p < 0)

{

p += 2 \* x + 3;

}

else

{

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

y--;

}

x++;

}

}

int main()

{

int gd = DETECT, gm;

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

int xc, yc, radius;

std::cout << "Enter the x-coordinate of the center: ";

std::cin >> xc;

std::cout << "Enter the y-coordinate of the center: ";

std::cin >> yc;

std::cout << "Enter the radius: ";

std::cin >> radius;

drawCircle(xc, yc, radius);

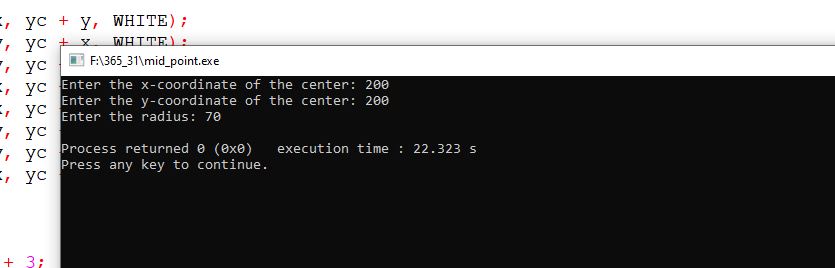
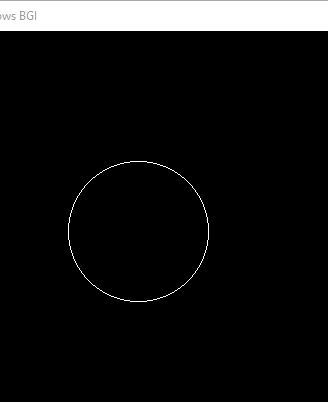
delay(5000);

closegraph();

return 0;

}

**Output:**

****

**Ellipse:**

#include <graphics.h>

#include <bits/stdc++.h>

using namespace std;

void drawEllipseScanConversion(int centerX, int centerY, int a, int b) {

int x = 0;

int y = b;

int aSq = a \* a;

int bSq = b \* b;

int twoASq = 2 \* aSq;

int twoBSq = 2 \* bSq;

int decision = bSq - aSq \* b + aSq / 4;

while (twoBSq \* x < twoASq \* y) {

putpixel(centerX + x, centerY + y, WHITE);

putpixel(centerX - x, centerY + y, WHITE);

putpixel(centerX + x, centerY - y, WHITE);

putpixel(centerX - x, centerY - y, WHITE);

if (decision <= 0) {

x++;

decision += twoBSq \* x + bSq;

} else {

x++;

y--;

decision += twoBSq \* x - twoASq \* y + bSq;

}

}

decision = bSq \* (x + 0.5) \* (x + 0.5) + aSq \* (y - 1) \* (y - 1) - aSq \* bSq;

while (y >= 0) {

putpixel(centerX + x, centerY + y, WHITE);

putpixel(centerX - x, centerY + y, WHITE);

putpixel(centerX + x, centerY - y, WHITE);

putpixel(centerX - x, centerY - y, WHITE);

if (decision > 0) {

y--;

decision += aSq - twoASq \* y;

} else {

y--;

x++;

decision += twoBSq \* x - twoASq \* y + aSq;

}

}

}

int main() {

int gd = DETECT, gm;

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

int centerX, centerY, a, b;

cout << "Enter center coordinates (x, y): ";

cin >> centerX >> centerY;

cout << "Enter semi-major axis (a): ";

cin >> a;

cout << "Enter semi-minor axis (b): ";

cin >> b;

drawEllipseScanConversion(centerX, centerY, a, b);

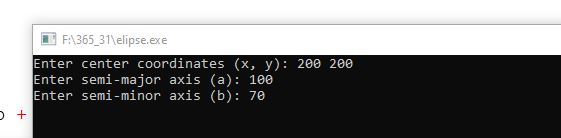
getch();

closegraph();

return 0;

}

**Output:**

****