



Jahangirnagar University
3rd Year 1st Semester Examination 2022

Course title: Computer Graphics Laboratory
Course code: CSE-304

Submitted to-

Dr. Mohammad Shorif Uddin
Professor
Department of Computer Science and Engineering
Jahangirnagar University

Dr. Morium Akter
Associate Professor
Department of Computer Science and Engineering
Jahangirnagar University

Submitted by:

Name : Akila Nipo
Class Roll : 368
Exam Roll : 202180

❖ Experiment No-1: Scan Conversion of a Point:

Source Code:

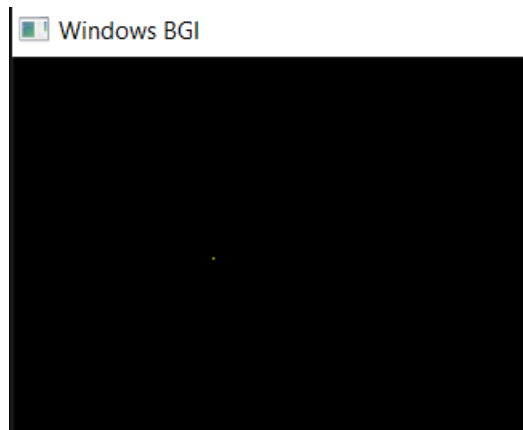
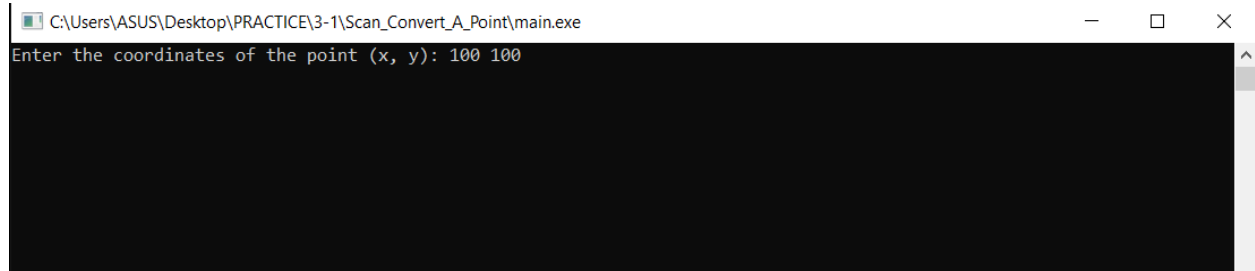
```
#include <graphics.h>
#include <iostream>

int main() {
    int gd = DETECT, gm;
    initgraph(&gd, &gm, "");

    int x, y;
    std::cout << "Enter the coordinates of the
point (x, y): ";
    std::cin >> x >> y;

    putpixel(x, y, WHITE);

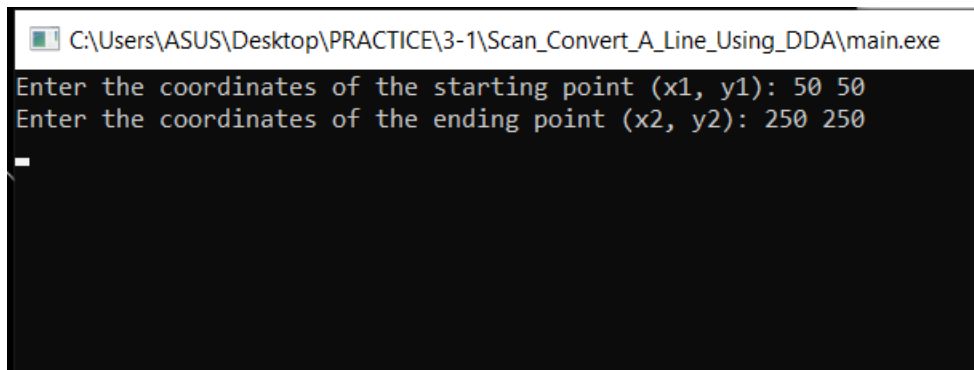
    delay(5000000);
    closegraph();
    return 0;
}
```

Screenshot:

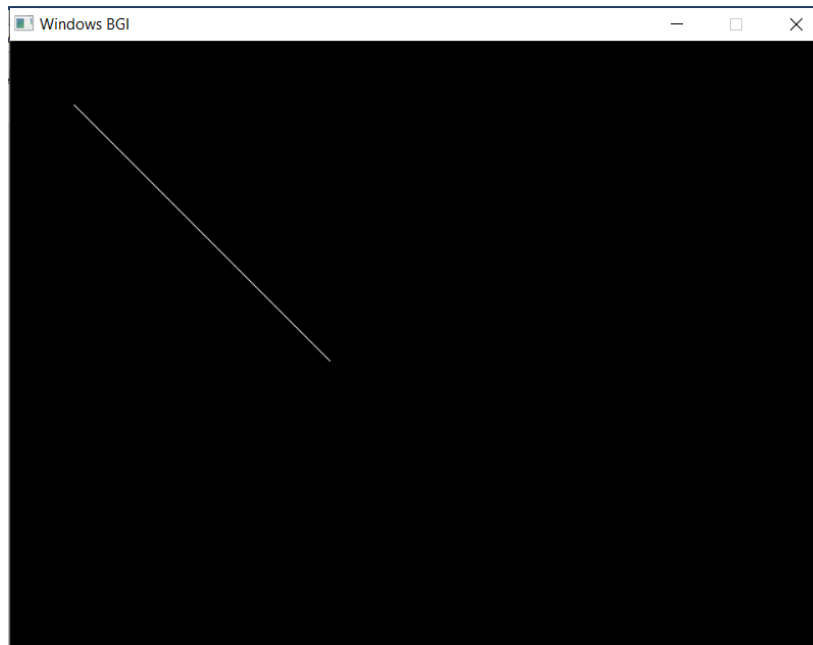
❖ Experiment No-2: Scan Conversion of a Line Using DDA Algorithm:

Source Code:

<pre>#include <graphics.h> #include <iostream> int main() { int gd = DETECT, gm; initgraph(&gd, &gm, ""); int x1, y1, x2, y2; std::cout << "Enter the coordinates of the starting point (x1, y1): "; std::cin >> x1 >> y1; std::cout << "Enter the coordinates of the ending point (x2, y2): "; std::cin >> x2 >> y2; int dx = x2 - x1; int dy = y2 - y1; int steps = std::max(abs(dx), abs(dy)); float xIncrement = static_cast<float>(dx) / steps;</pre>	<pre>float yIncrement = static_cast<float>(dy) / steps; float x = x1; float y = y1; for (int i = 0; i <= steps; i++) { putpixel(static_cast<int>(x), static_cast<int>(y), WHITE); x += xIncrement; y += yIncrement; } delay(5000000); closegraph(); return 0; }</pre>
--	---

Screenshot:

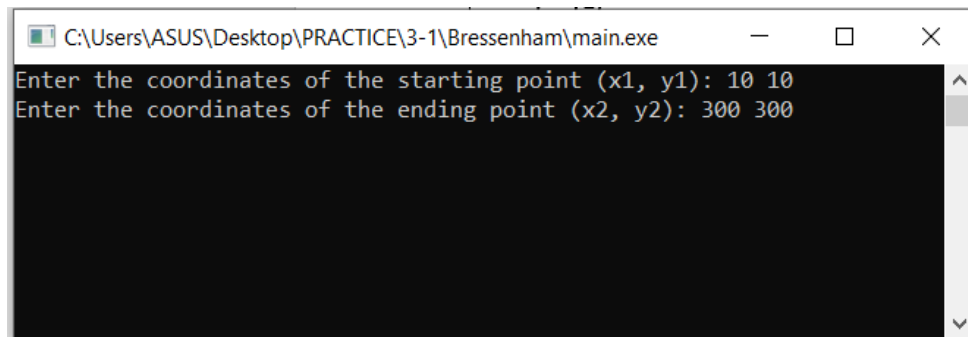
```
C:\Users\ASUS\Desktop\PRACTICE\3-1\Scan_Convert_A_Line_Using_DDA\main.exe
Enter the coordinates of the starting point (x1, y1): 50 50
Enter the coordinates of the ending point (x2, y2): 250 250
_
```



❖ Experiment No-3: Scan Conversion of a Line Using Bresenham's Algorithm:

Source Code:

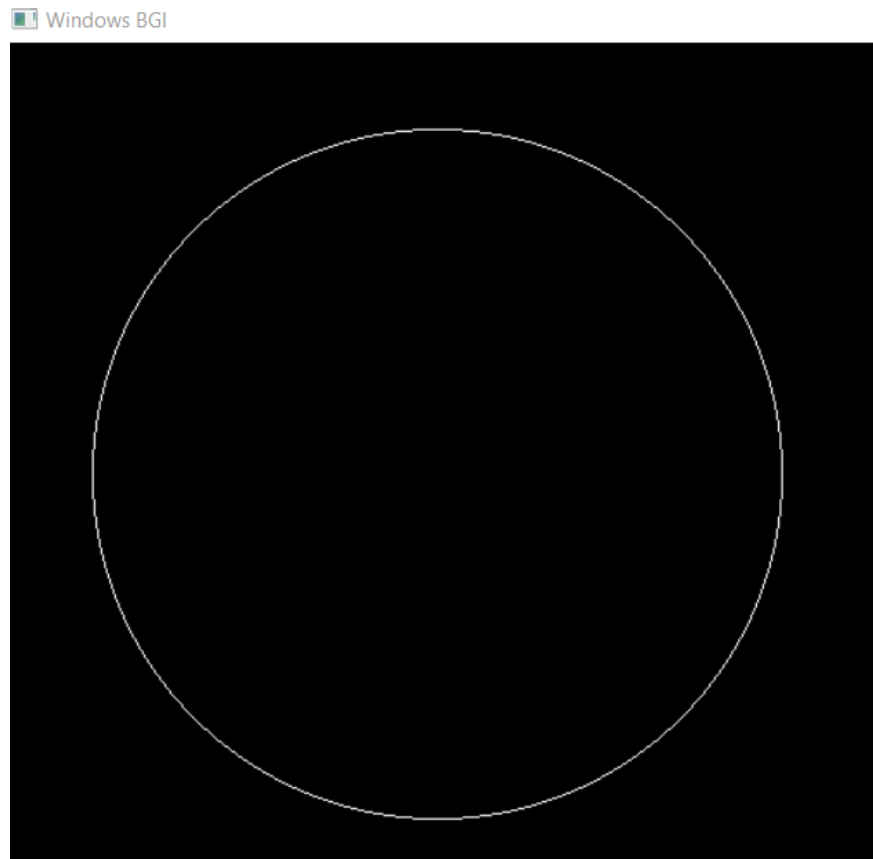
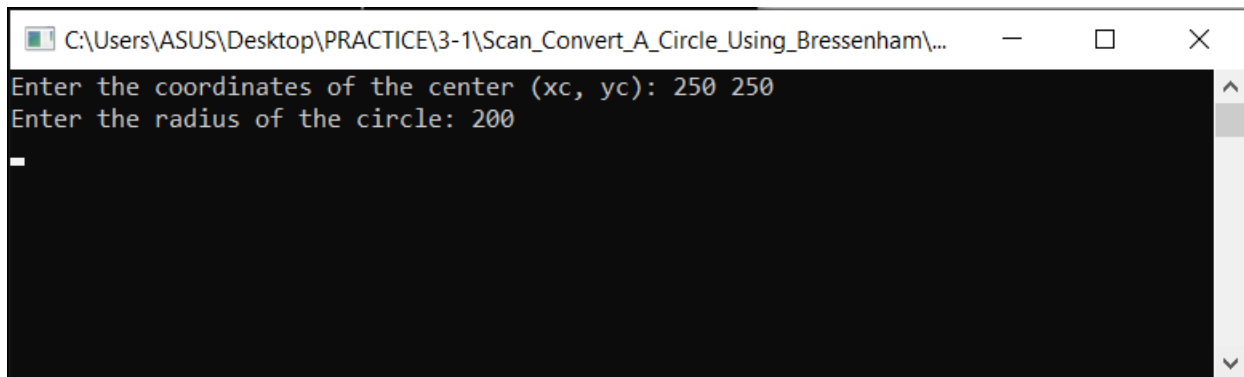
<pre>#include <graphics.h> #include <iostream> int main() { int gd = DETECT, gm; initgraph(&gd, &gm, ""); int x1, y1, x2, y2; std::cout << "Enter the coordinates of the starting point (x1, y1): "; std::cin >> x1 >> y1; std::cout << "Enter the coordinates of the ending point (x2, y2): "; std::cin >> x2 >> y2; int dx = abs(x2 - x1); int dy = abs(y2 - y1); int x, y; if (x1 < x2) { x = x1; y = y1; } </pre>	<pre>else { x = x2; y = y2; x2 = x1; y2 = y1; } int p = 2 * dy - dx; putpixel(x, y, WHITE); while (x < x2) { x++; if (p < 0) { p += 2 * dy; } else { y++; p += 2 * (dy - dx); } putpixel(x, y, WHITE); } delay(5000*3600); closegraph(); return 0; </pre>
--	--

Screenshot:

❖ Experiment No-4: Scan Conversion of a Circle Using Bresenham's Circle Algorithm:

Source Code:

<pre>#include <graphics.h> #include <iostream> int main() { int gd = DETECT, gm; initgraph(&gd, &gm, ""); int xc, yc, radius; std::cout << "Enter the coordinates of the center (xc, yc): "; std::cin >> xc >> yc; std::cout << "Enter the radius of the circle: "; std::cin >> radius; int x = 0; int y = radius; int d = 3 - 2 * radius; while (x <= y) { putpixel(xc + x, yc + y, WHITE); putpixel(xc - x, yc + y, WHITE); putpixel(xc + x, yc - y, WHITE); putpixel(xc - x, yc - y, WHITE);</pre>	<pre> 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 += 4 * x + 6; } else { d += 4 * (x - y) + 10; y--; } x++; } delay(500000); closegraph(); return 0; }</pre>
--	---

Screenshot:

— THE END —