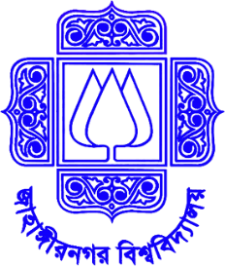
**Lab Report. 01**

*Course title: Computer Graphics Lab*

*Course code: CSE-304*

*3rd Year 1st Semester 2022*

**Date of Submission**: 28-05-2023



###### **Submitted to-**

###### Dr. Mohammad Shorif Uddin

###### Professor

*Department of Computer Science and Engineering*

*Jahangirnagar University*

*Savar, Dhaka-1342*

*And*

*Dr. Morium Akter*

###### Associate Professor

*Department of Computer Science and Engineering*

*Jahangirnagar University*

*Savar, Dhaka-1342*

| **Sl** | **Class Roll** | **Exam Roll** | **Name** |
| --- | --- | --- | --- |
| 01 | 401 | 202213 | Md. Niaz Rahaman |

**1.Scan conversion of a point**

**Sourcecode:**

#include <bits/stdc++.h>

#include<graphics.h>

int main()

{

int gd = DETECT, gm;

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

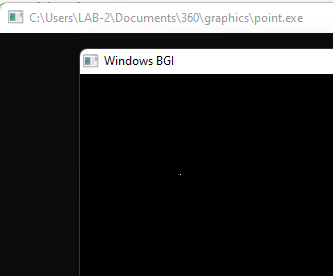
putpixel(100,100,WHITE);

getch();

closegraph();

}

**Output:**



**2.Scan conversion of a line using DDA Algorithm**

**Sourcecode:**

#include <bits/stdc++.h>

#include <graphics.h>

int main()

{

int gd = DETECT, gm;

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

int x1 = 100, y1 = 100;

int x2 = 500, y2 = 300;

int dx = x2 - x1;

int dy = y2 - y1;

int steps = abs(dx) > abs(dy) ? abs(dx) : abs(dy);

float xInc = dx / (float)steps;

float yInc = dy / (float)steps;

float x = x1;

float y = y1;

for (int i = 0; i <= steps; i++)

{

putpixel(x, y, GREEN);

x += xInc;

y += yInc;

}

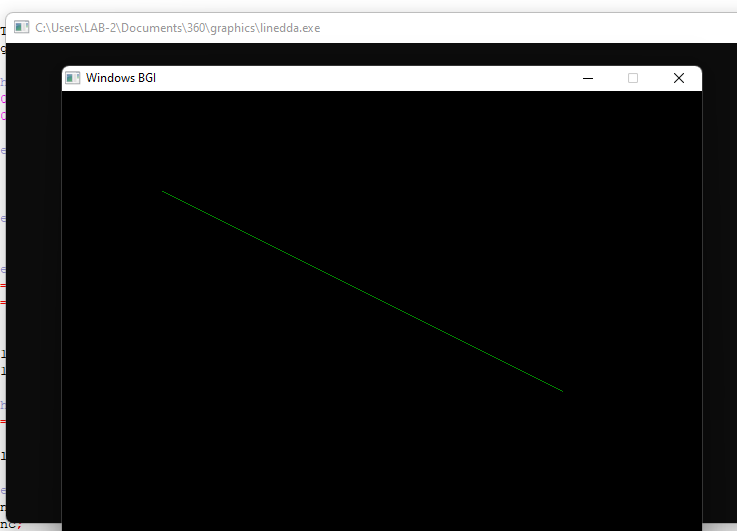
getch();

closegraph();

return 0;

}

**Output:**



**3.Scan conversion of a line using Bresenham Algorithm:**

**Sourcecode:**

#include <bits/stdc++.h>

#include <graphics.h>

void drawLine(int x1, int y1, int x2, int y2)

{

int dx = abs(x2 - x1);

int dy = abs(y2 - y1);

int sx = (x1 < x2) ? 1 : -1;

int sy = (y1 < y2) ? 1 : -1;

int err = dx - dy;

while (true)

{

putpixel(x1, y1, GREEN);

if (x1 == x2 && y1 == y2)

break;

int e2 = 2 \* err;

if (e2 > -dy) {

err -= dy;

x1 += sx;

if (e2 < dx)

{

err += dx;

y1 += sy;

}

}

}

int main()

{

int gd = DETECT, gm;

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

int x1 = 100, y1 = 100;

int x2 = 500, y2 = 300;

drawLine(x1, y1, x2, y2);

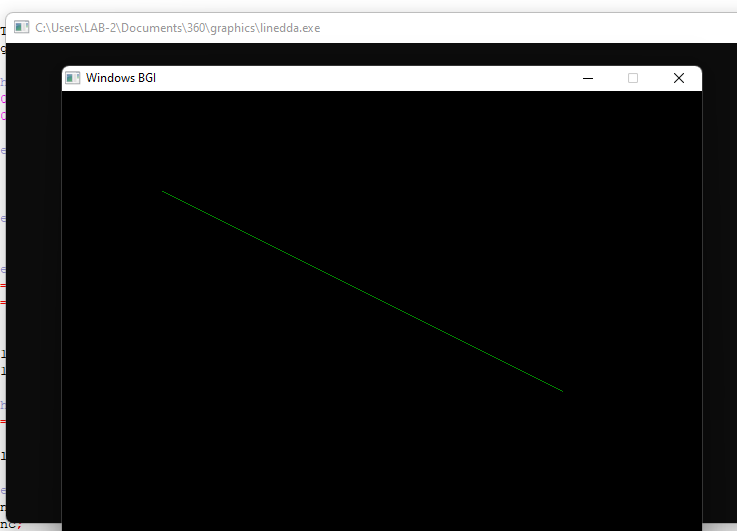
getch();

closegraph();

return 0;

}

**Output:**



**4.Scan conversion of circle using Bresenham Algorithm:**

**Sourcecode:**

#include <bits/stdc++.h>

#include <graphics.h>

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

{

int x = 0;

int y = radius;

int d = 3 - 2 \* radius;

while (x <= y)

{

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

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

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

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

else

{

d += 4 \* (x - y) + 10;

y--;

}

x++;

}

}

int main()

{

int gd = DETECT, gm;

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

int xc = 200, yc = 200;

int radius = 200;

drawCircle(xc, yc, radius);

getch();

closegraph();

return 0;

}

**Output:**

