**LAB Assignment-01**

*Course title: Computer Graphics Laboratory*

*Course code: CSE-304*

*3nd  Year 1st Semester Examination 2022*

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

****

**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 | **385** |  | **Md.Soad Anam** |

**Experiment No. : 01**

**Experiment Title : Scan conversion of a point**

**Source Code:**

#include<bits/stdc++.h>

#include <graphics.h>

using namespace std;

int main()

{

int gd = DETECT, gm;

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

setbkcolor(WHITE);

float xi,yi;

int w,h,i,j;

cout<<"Enter co-ordinates the point for scan conversion: ";

cin>>xi>>yi;

int x,y;

x=round(xi);

y=round(yi);

putpixel(x, y, RED);

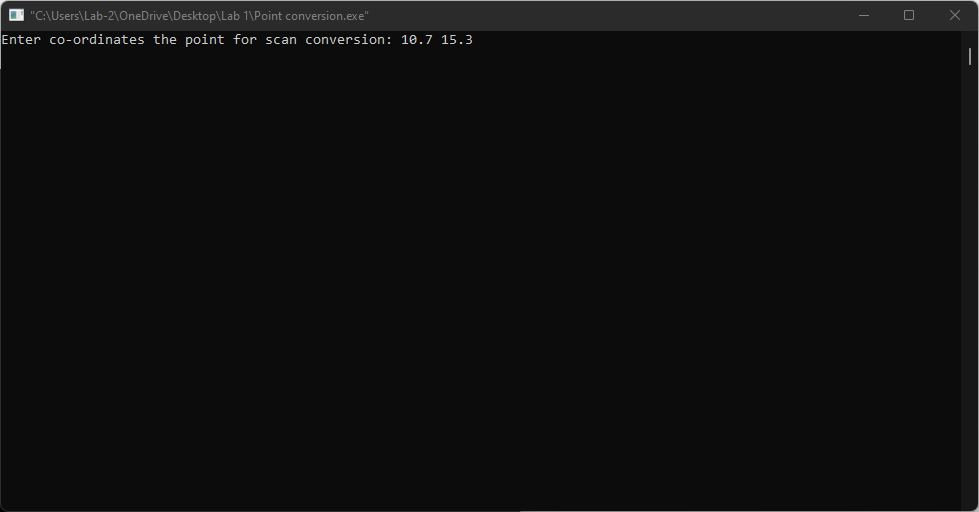
getch();

closegraph();

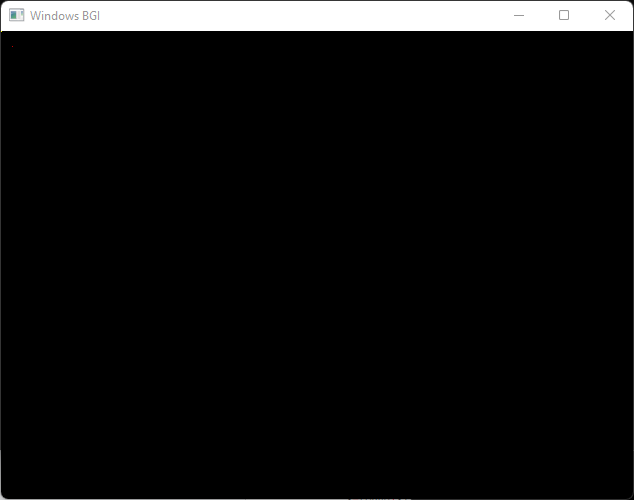
return 0;

}

**Output:**

****

**Fig-01: The point as input**

****

**Fig-02: The converted point in the image**

**Experiment No. : 02**

**Experiment Title : Scan conversion of a line using DDA algorithm**

**Source Code:**

#include<bits/stdc++.h>

#include <graphics.h>

using namespace std;

int main()

{

int x0,y0,x1,y1;

int m,c,i;

float x,y,dx,dy,s;

cout<<"Enter the staring point: ";

cin>>x0>>y0;

cout<<"Enter the ending point: ";

cin>>x1>>y1;

int gd = DETECT, gm;

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

setbkcolor(WHITE);

dx=(float)x1-x0;

dy=(float)y1-y0;

s = max(abs(dx), abs(dy));

dx = dx/s;

dy = dy/s;

x = x0;

y = y0;

i = 1;

while(i<= s)

{

putpixel(x, y, BLUE);

x += dx;

y += dy;

i=i+1;

}

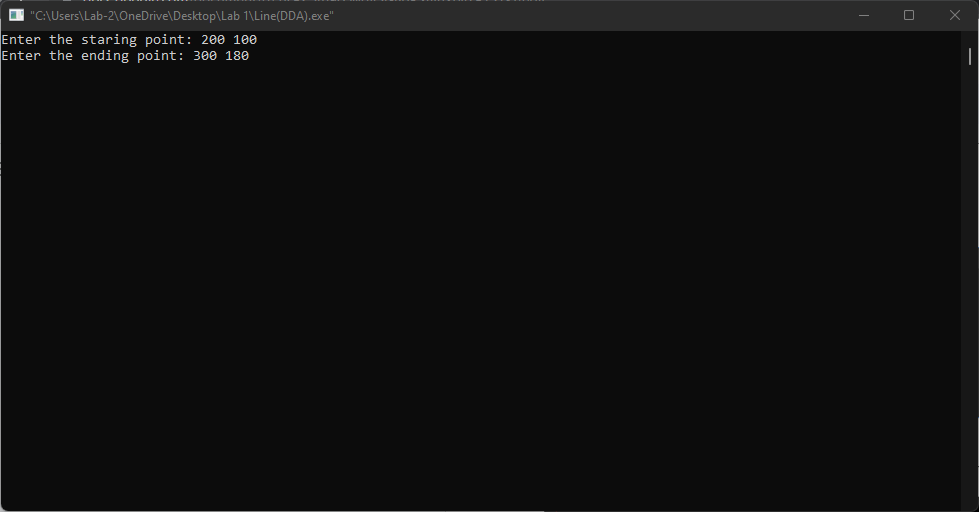
getch();

closegraph();

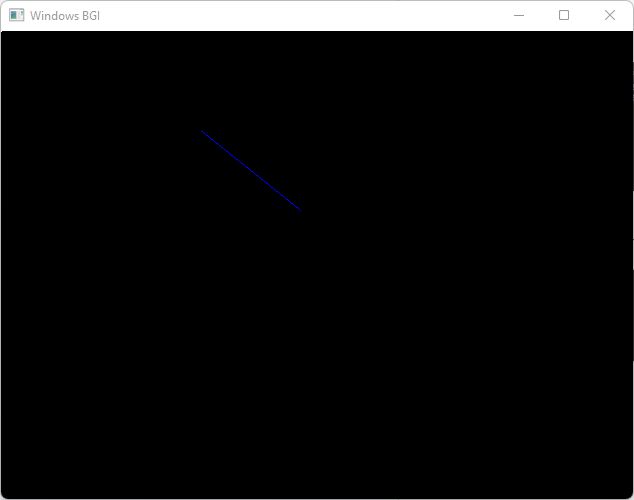
return 0;

}

**Output:**

****

**Fig-01: The points as input**

****

**Fig-02: The converted points of line**

**Experiment No. : 03**

**Experiment Title : Scan conversion of a line using Bresenham’s algorithm**

**Source Code:**

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

int main()

{

float x, y,dx,dy,step;

int x0, x1, y0, y1,i,p;

int gd = DETECT, gm;

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

setbkcolor(WHITE);

cout<<"Enter co-ordinates for first point: ";

cin>>x0>>y0;

cout<<"Enter co-ordinates for second point: ";

cin>>x1>>y1;

dx = (float)(x1 - x0);

dy = (float)(y1 - y0);

p=(2\*dy)-dx;

x = x0;

y = y0;

i = 1;

while(x<x1)

{

if(p>=0)

{

putpixel(x,y,7);

y=y+1;

p=p+2\*dy-2\*dx;

}

else

{

putpixel(x,y,7);

p=p+2\*dy;}

x=x+1;

}

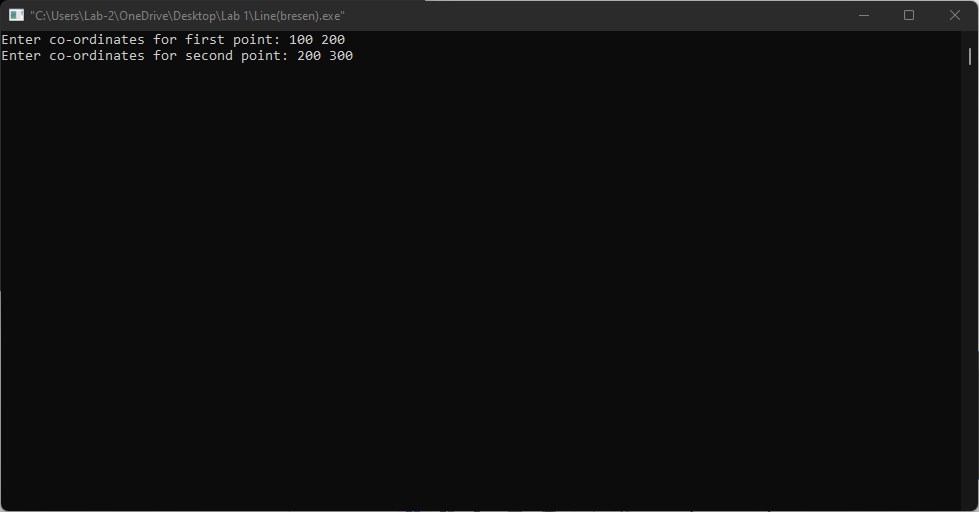
getch();

closegraph();

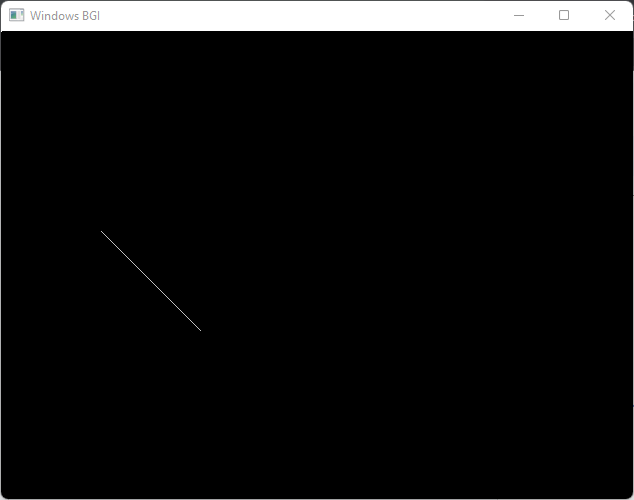
return 0;

}

**Output:**

****

**Fig-01: The points as input**

****

**Fig-02: The converted points of line**

**Experiment No. : 04**

**Experiment Title : Scan conversion of a circle using Bresenham’s algorithm**

**Source Code:**

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

int main()

{

int gd = DETECT, gm;

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

setbkcolor(WHITE);

int x,y,r;

cout << "Enter the coordinates of the center: ";

cin >>x>>y ;

cout << "Enter the radius of the circle: ";

cin >> r;

int xi,yi,d;

xi = 0;

yi = r;

d = 3 - 2 \* r;

while (xi <= yi)

{

putpixel(x + xi, y + yi, BLUE);

putpixel(x - xi, y + yi, BLUE);

putpixel(x + xi, y - yi, BLUE);

putpixel(x - xi, y - yi, BLUE);

putpixel(x + yi, y + xi, BLUE);

putpixel(x - yi, y + xi, BLUE);

putpixel(x + yi, y - xi, BLUE);

putpixel(x - yi, y - xi, BLUE);

if (d <= 0)

{

d += 4 \* xi + 6;

}

else

{

d += 4 \* (xi - yi) + 10;

yi--;

}

xi++;

}

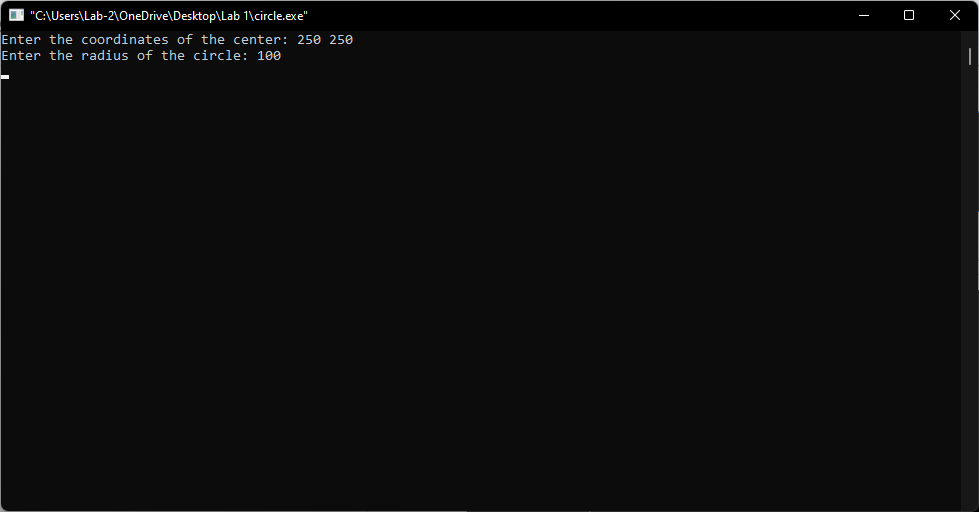
getch();

closegraph();

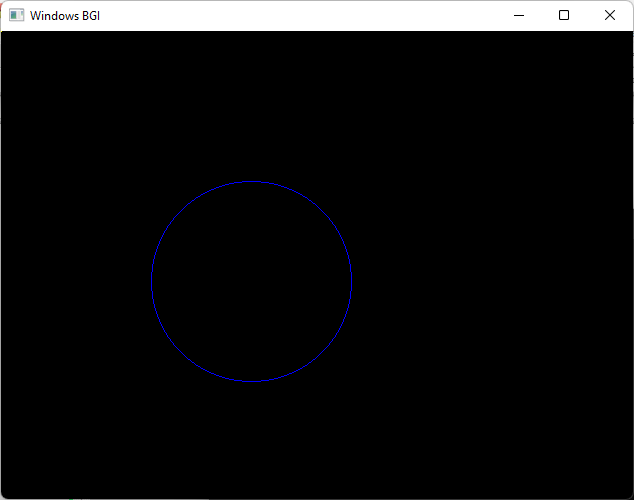
return 0;

}

**Output:**

****

**Fig-01: The points as input**

****

**Fig-02: The converted points of circle**