**Lab Report 1**

*Course title: Computer Graphics Laboratory*

*Course code: CSE-304*

*3rd Year 1st Semester*

**Date of Submission**: 28/5/2023

A picture containing text

Description automatically generated

**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 | 384 |  | Taufiq Islam |

Department of Computer Science and Engineering

Jahangirnagar University

Savar, Dhaka, Bangladesh

# Scan a point:

Source Code:

#include<bits/stdc++.h>

#include<graphics.h>

int main()

{

float x,y;

printf("Enter point(x,y): ");

scanf("%f %f",&x,&y);

int sx,sy;

sx = floor(x);

sy = floor(y);

int gd= DETECT, gm;

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

putpixel(sx,sy,WHITE);

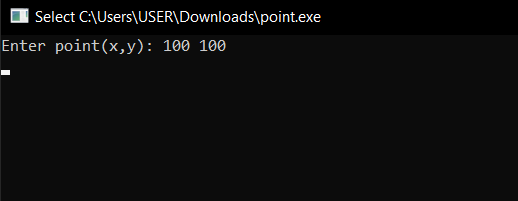
getch();

closegraph();

return 0;

}

Output:



A screenshot of a computer

Description automatically generated

## Scanning a line using DDA algorithm:

Source Code:

#include<bits/stdc++.h>

#include<graphics.h>

int main()

{

float x1,y1,x2,y2,m1,b1,m,b;

int sx1,sy1,sx2,sy2;

printf("Enter first point(x1,y1): ");

scanf("%f %f",&x1,&y1);

sx1 = floor(x1);

sy1 = floor(y1);

printf("Enter second point(x2,y2): ");

scanf("%f %f",&x2,&y2);

sx2 = floor(x2);

sy2 = floor(y2);

m = (sy2-sy1)/(sx2-sx1);

b = sy1 - (m \* sx1);

int steps,k;

float x\_inc,y\_inc;

if(abs(sx2-sx1) > abs(sy2-sy1))

{

steps=abs(sx2-sx1);

}

else

{

steps=abs(sy2-sy1);

}

x\_inc = (sx2-sx1) / (float)steps;

y\_inc = (sy2-sy1) / (float)steps;

int gd= DETECT, gm;

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

putpixel(sx1,sy1,WHITE);

for(k = 0;k < steps;k++)

{

sx1 += x\_inc;

sy1 += y\_inc;

putpixel(round(sx1),round(sy1),WHITE);

}

getch();

closegraph();

return 0;

}

Output Screenshot:

A screenshot of a computer program

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

# Scanning a line using bresenham line algorithm:

Source Code:

#include<bits/stdc++.h>

#include<graphics.h>

int main()

{

float x1,y1,x2,y2,m1,b1,m,b;

int sx1,sy1,sx2,sy2;

printf("Enter first point(x1,y1): ");

scanf("%f %f",&x1,&y1);

sx1 = floor(x1);

sy1 = floor(y1);

printf("Enter second point(x2,y2): ");

scanf("%f %f",&x2,&y2);

sx2 = floor(x2);

sy2 = floor(y2);

int steps,k,dx,dy;

float x\_inc,y\_inc;

dx = abs(sx2-sx1);

dy = abs(sy2-sy1);

int err = dx-dy;

if(sx1 < sx2)

{

x\_inc = 1;

}

else

{

x\_inc = -1;

}

if(sy1 < sy2)

{

y\_inc = 1;

}

else

{

y\_inc = -1;

}

int gd= DETECT, gm;

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

while(1)

{

putpixel(sx1,sy1,WHITE);

if(sx1 == sx2 && sy1 == sy2)

{

break;

}

int newerr = 2 \* err;

if(newerr > -dy)

{

err -= dy;

sx1 += x\_inc;

}

if(newerr < dx)

{

err += dx;

sy1 += y\_inc;

}

}

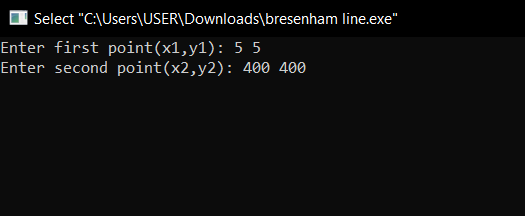
getch();

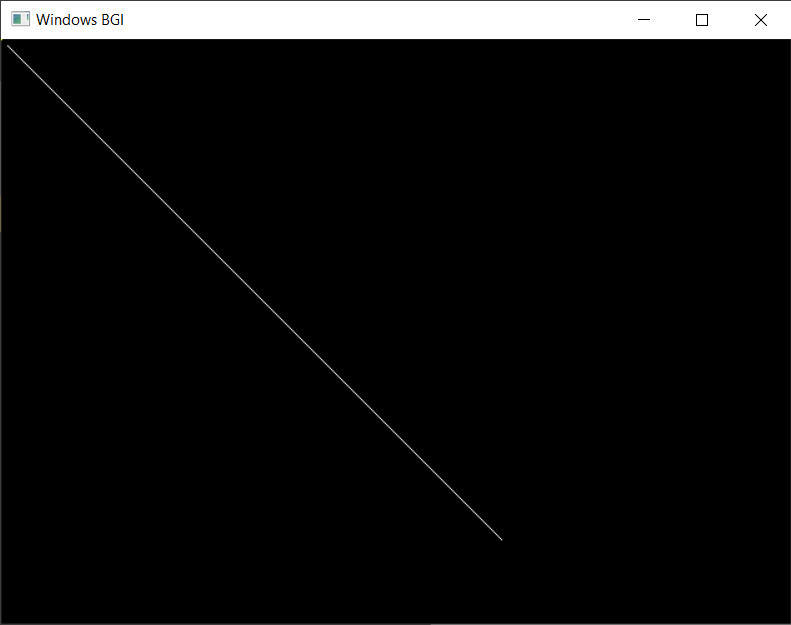
closegraph();

return 0;

}

Output:





# Scanning a circle using bresenham circle algorithm:

Source Code:

#include<bits/stdc++.h>

#include<graphics.h>

int main()

{

float x,y,r;

int sx,sy,sr;

printf("Enter center point(x,y): ");

scanf("%f %f",&x,&y);

sx = floor(x);

sy = floor(y);

printf("Enter radius: ");

scanf("%f",&r);

sr = floor(r);

int dx = 0;

int dy = sr;

int d = 3 - (2 \* sr);

int gd= DETECT, gm;

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

while(dx <= dy)

{

putpixel(sx+dx,sy+dy,WHITE);

putpixel(sx-dx,sy+dy,WHITE);

putpixel(sx+dx,sy-dy,WHITE);

putpixel(sx-dx,sy-dy,WHITE);

putpixel(sx+dy,sy+dx,WHITE);

putpixel(sx-dy,sy+dx,WHITE);

putpixel(sx+dy,sy-dx,WHITE);

putpixel(sx-dy,sy-dx,WHITE);

if(d < 0)

{

d += (4\*dx) + 6;

}

else

{

d += (4 \* (dx - dy)) + 10;

dy--;

}

dx++;

}

getch();

closegraph();

return 0;

}

Output:

