**Lab report - 01**

*Course title: Computer Graphics Laboratory*

*Course code: CSE-304*

*3rd Year 1st Semester 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*

|  |  |  |
| --- | --- | --- |
| **Name** | **Class roll** | **Exam roll** |
| Hasneen Tamanna Totinee | 362 | 202174 |

**1. Scan conversion of a point**

Code:

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

int main()

{

float x1,x2;

cout<<"point(x,y) to be scanned:"<<endl;

cin>>x1 >>x2;

int a,b;

a=floor(x1);

b=floor(x2);

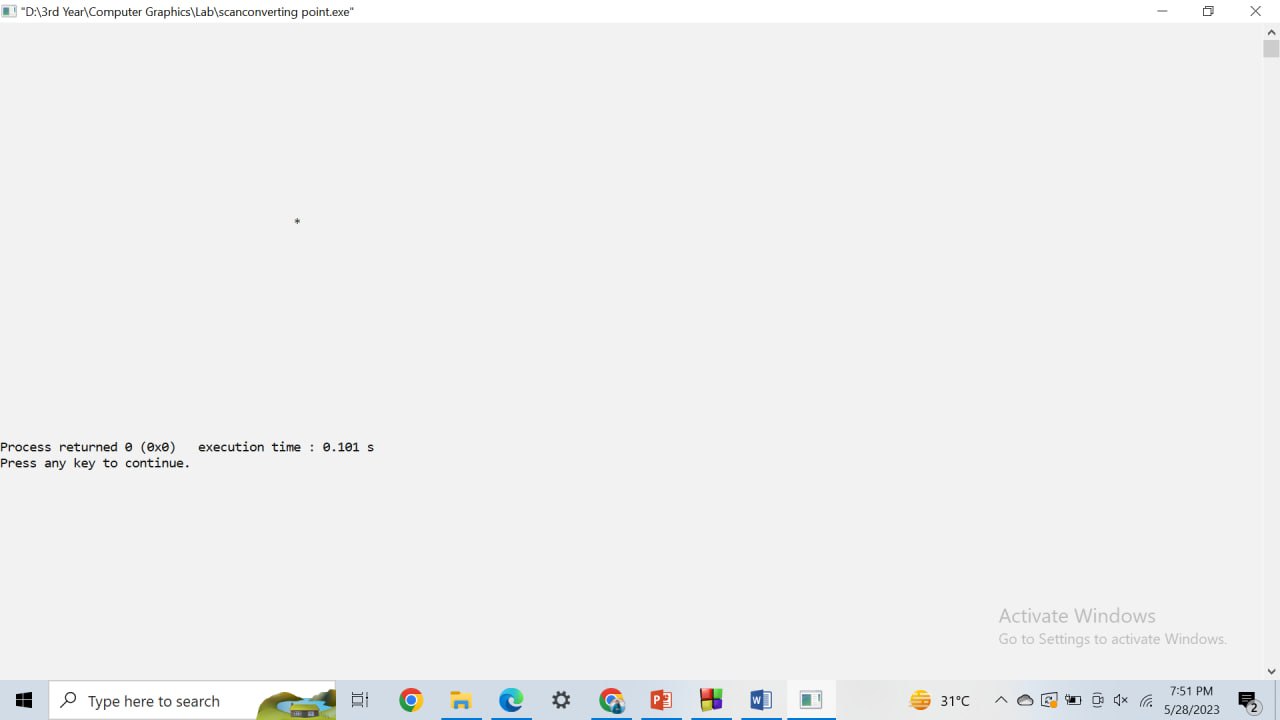
cout<<"after scan conversion:"<<endl;

cout<<a <<" "<< b<<endl;

return 0;

}

Output:



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

Code:

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

void drawLine(int x0,int y0,int x1,int y1)

{

int dx=x1-x0;

int dy=y1-y0;

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

float xinc= static\_cast<float>(dx)/step;

float yinc= static\_cast<float>(dy)/step;

float x=x0;

float y=y0;

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

{

//cout<<"("<<round(x)<<","<< round(y)<<")"<< endl;

putpixel(static\_cast<int>(x),static\_cast<int>(y),WHITE);

x+=xinc;

y+=yinc;

}

}

int main()

{

int gd=DETECT,gm;

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

int x0,y0,x1,y1;

cout<<"starting point : ";

cin>>x0>>y0;

cout<<"ending point : ";

cin>>x1>>y1;

//cout<<"line coordinates: "<<endl;

drawLine(x0,y0,x1,y1);

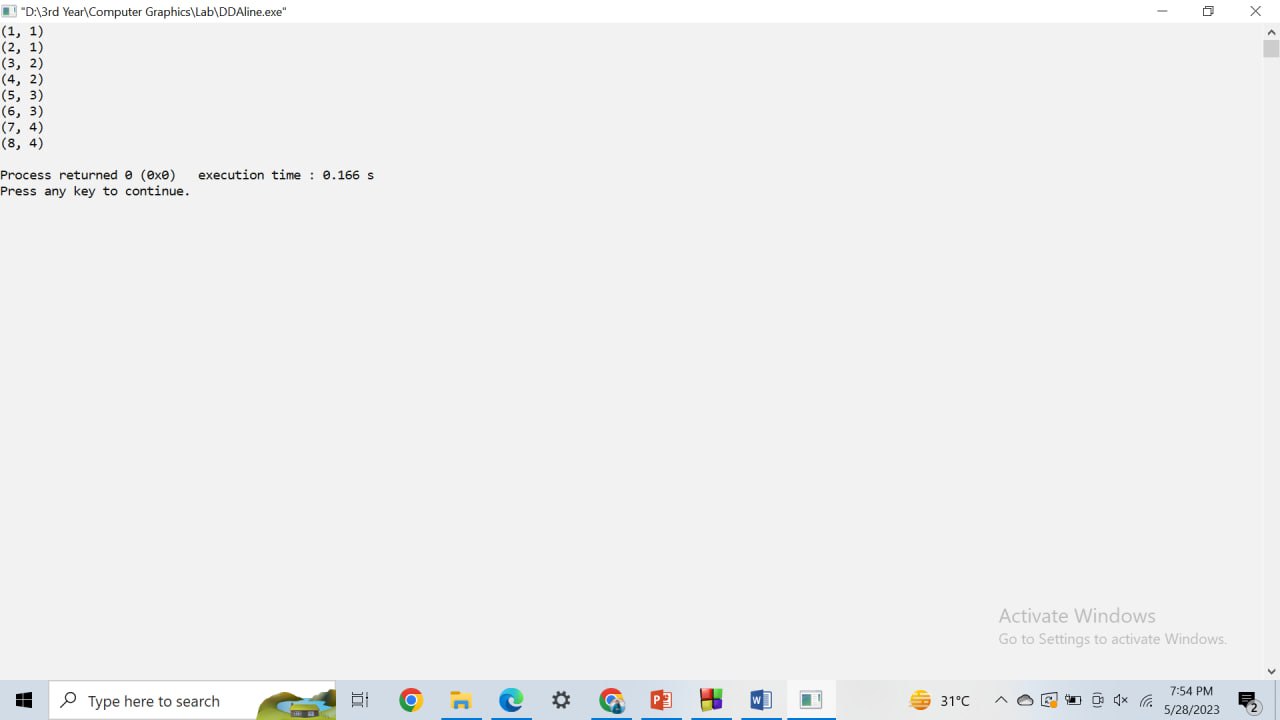
delay(50000000);

closegraph();

return 0;

}

Output:



**3. Scan conversion of a line using Bresenham’s algorithm**

Code:

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

void drawLine(int x0,int y0,int x1,int y1)

{

int dx=x1-x0;

int dy=y1-y0;

int sx=(x0<x1)? 1:-1;

int sy=(y0<y1)? 1:-1;

int errorr= dx-dy;

// int x=x1;

// int y=y1;

while(true)

{

putpixel(x0,y0,WHITE);

if(x0==x1 && y0==y1) break;

int dt= 2\*errorr;

if(dt>-dy)

{

errorr-=dy;

x0+=sx;

}

if(dt<dx)

{

errorr+=dx;

y0+=sy;

}

}

}

int main()

{

int gd=DETECT,gm;

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

int x0,y0,x1,y1;

cout<<"starting point : ";

cin>>x0>>y0;

cout<<"ending point : ";

cin>>x1>>y1;

//cout<<"line coordinates: "<<endl;

drawLine(x0,y0,x1,y1);

delay(50000000);

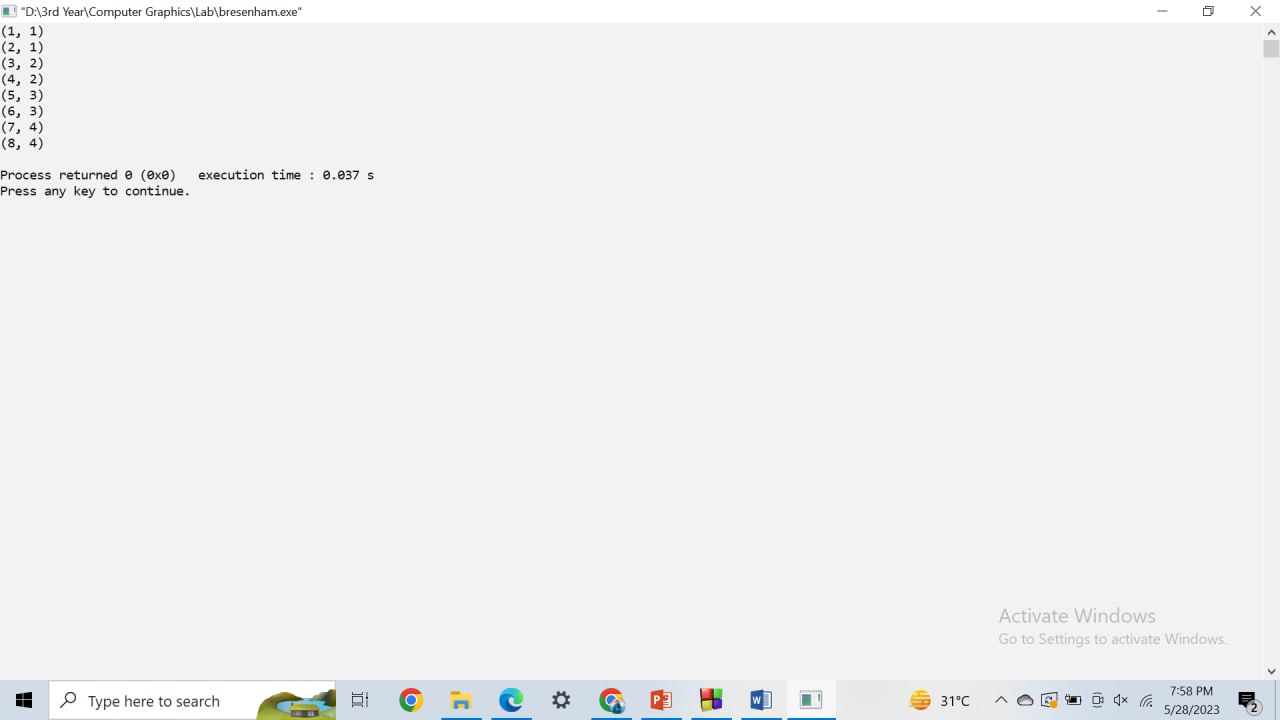
closegraph();

return 0;

return 0;

}

Output:



**4. Scan conversion of a circle using Bresenham’s circle algorithm**

Code:

#include<bits/stdc++.h>

#include<graphics.h>

using namespace std;

void drawLine(int x0,int y0,int x1,int y1)

{

int dx=x1-x0;

int dy=y1-y0;

int sx=(x0<x1)? 1:-1;

int sy=(y0<y1)? 1:-1;

int errorr= dx-dy;

// int x=x1;

// int y=y1;

while(true)

{

putpixel(x0,y0,WHITE);

if(x0==x1 && y0==y1) break;

int dt= 2\*errorr;

if(dt>-dy)

{

errorr-=dy;

x0+=sx;

}

if(dt<dx)

{

errorr+=dx;

y0+=sy;

}

}

}

int main()

{

int gd=DETECT,gm;

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

int x0,y0,x1,y1;

cout<<"starting point : ";

cin>>x0>>y0;

cout<<"ending point : ";

cin>>x1>>y1;

//cout<<"line coordinates: "<<endl;

drawLine(x0,y0,x1,y1);

delay(50000000);

closegraph();

return 0;

return 0;

}

Output:

