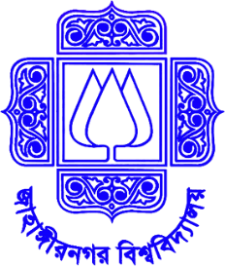
**Lab Report. 01**

*Course title: Computer Graphics Laboratory*

*Course code: CSE-304*

*3rd Year 1st Semester 2022*

**Date of Submission**: 28 May,2023



***Submitted to-***

*Dr. Mohammad*

*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*

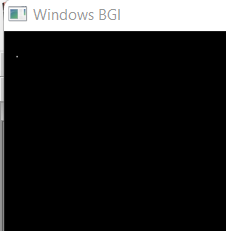
| Name | Roll |
| --- | --- |
| Sunirmol Mollik | 386 |

**Labwork:01**  Scan Conversion of a point.

**Source Code:**

| #include<graphics.h>  #include<conio.h>  #include<bits/stdc++.h>  using namespace std;  int main()  {     int gd = DETECT, gm; | initgraph(&gd, &gm, "C:\\TC\\BGI");  float x = 10, y = 20;          putpixel (x, y, RED);     getch();     closegraph();     return 0;  } |
| --- | --- |

**Output:**

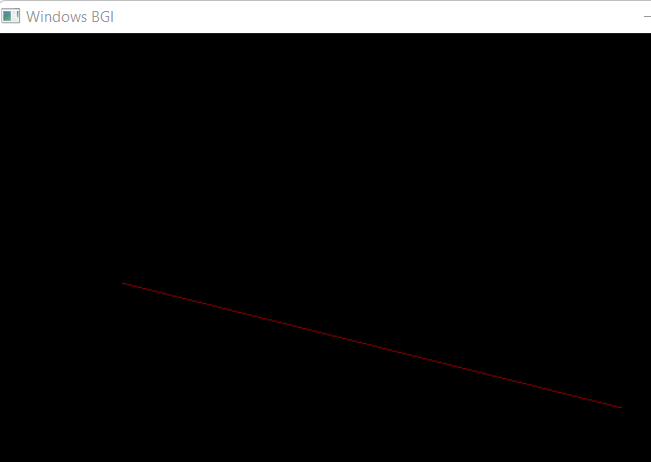


**Labwork:02**  Scan conversion of a line using DDA algorithm.

**Source Code:**

| #include<graphics.h>  #include<conio.h>  #include<stdio.h>  int main()  {      int gd = DETECT ,gm, i;      float x, y,dx,dy,steps;      int x0, x1, y0, y1;      initgraph(&gd, &gm, "C:\\TC\\BGI");      setbkcolor(WHITE);      x0 = 100 , y0 = 200, x1 = 500, y1 = 300;      dx = (float)(x1 - x0);  dy = (float)(y1 - y0);      if(dx>=dy)             {          steps = dx;  }  else    {          steps = dy; | }      dx = dx/steps;      dy = dy/steps;      x = x0;      y = y0;      i = 1;      while(i<= steps)      {          putpixel(x, y, RED);          x += dx;          y += dy;          i=i+1;      }      getch();      closegraph();  } |
| --- | --- |

**Output:**

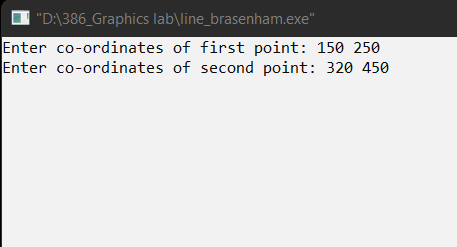


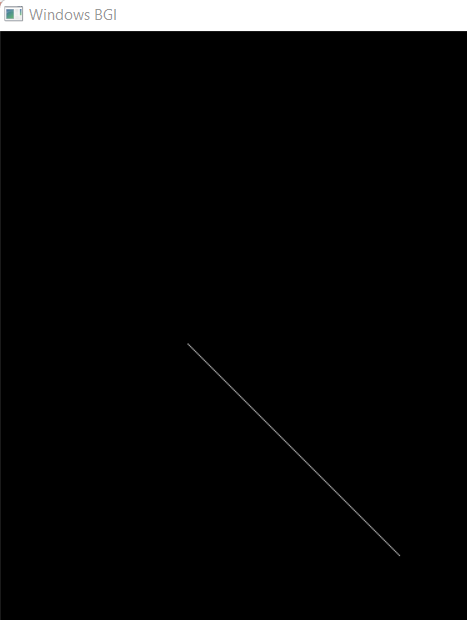
**Labwork:03**  Scan Conversion of a line using Brasenham Algorithm.

**Source Code:**

| #include<stdio.h>  #include<graphics.h>  #include<bits/stdc++.h>  void drawline(int x0, int y0, int x1, int y1)  {      int dx, dy, p, x, y;      dx=x1-x0;      dy=y1-y0;      x=x0;      y=y0;      p=2\*dy-dx;      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;          }  }  int main()  {      int gd = DETECT, gm;     initgraph(&gd, &gm, "C:\\TC\\BGI");      int error, x0, y0, x1, y1;      printf("Enter co-ordinates of first point: ");      scanf("%d%d", &x0, &y0);      printf("Enter co-ordinates of second point: ");      scanf("%d%d", &x1, &y1);      drawline(x0, y0, x1, y1);      getch();      return 0;  } |
| --- | --- |

**Output:**





**Labwork:04** Scan Conversion of a Circle using Brasenham Algorithm.

**Source Code:**

| #include <graphics.h>  #include <stdlib.h>  #include <stdio.h>  #include <conio.h>  #include <math.h>  void    EightWaySymmetricPlot(int xc,int yc,int x,int y)  {      putpixel(x+xc,y+yc,RED);      putpixel(x+xc,-y+yc,RED);      putpixel(-x+xc,-y+yc,RED);      putpixel(-x+xc,y+yc,RED);      putpixel(y+xc,x+yc,RED);      putpixel(y+xc,-x+yc,RED);      putpixel(-y+xc,-x+yc,RED);      putpixel(-y+xc,x+yc,RED);  }  void BresenhamCircle(int xc,int yc,int r)  {      int x=0,y=r,d=3-(2\*r);      EightWaySymmetricPlot(xc,yc,x,y);      while(x<=y)      {   if(d<=0)          {              d=d+(4\*x)+6;          }          else          {              d=d+(4\*x)-(4\*y)+10;              y=y-1;          }          x=x+1; | EightWaySymmetricPlot(xc,yc,x,y);      }  }  int  main(void)  {      int xc,yc,r,gdriver = DETECT, gmode, errorcode;      initgraph(&gdriver, &gmode, "C:\\TURBOC3\\BGI");      errorcode = graphresult();      if (errorcode != grOk)      {          printf("Graphics error: %s\n", grapherrormsg(errorcode));          printf("Press any key to halt:");          getch();          exit(1);      }      printf("Enter the values of xc and yc : ");      scanf("%d%d",&xc,&yc);      printf("Enter the value of radius: ");      scanf("%d",&r);      BresenhamCircle(xc,yc,r);      getch();      closegraph();      return 0;  } |
| --- | --- |

**Output:**

