

Lab Manual

CS602 – Computer Graphics



Prepared by
Muhammad Umar Farooq, Instructor CS

Week No.	Lab Topic	Page No.
1	Lab 1: Write a C Program that can write a pixel of red color at the 20th row and 20th column position.	2
2	Lab 2: Write the C program that will write draw the line using incremental line algorithm starting at the pixel position from 25th row and 30th column.	3
	Lab 3: Write the C program for drawing line by using Digital Differential Analyzer (DDA) Algorithm, straight line start on the screen starting at pixel position 30th row and 30th column.	4
3	Lab 4: Write the C program for drawing straight line by using Bresenham's line drawing algorithm the line start from the at pixel position 50th row and 60th column.	6
	Lab 5: Write the C program to draw circle at the following pixel position at 45th row and 55th column by using Cartesian coordinate system.	8
	Lab 6: Write the C program to draw circle at the following pixel position at 55th row and 65th column by using polar coordinate system.	9

Lab 1

Write a C Program that can write a pixel of red color at the 20th row and 20th column position.

```
#include <stdio.h>
#include <conio.h>
#include <graphics.h>
int main(){
    int gdriver, gmode;
    int errorcode;
    initgraph(&gdriver, &gmode, "");
    /* read result of initialization */
    errorcode = graphresult();
    if (errorcode != grOk)
    /* an error occurred */
    {
    printf("Graphics error: %s\n", getch()); exit(1);
    /* return with error code */
    }
    /* draw a pixel on 10th row and 10 column */
```

```
putpixel(60, 60, RED);
/* clean up */
closegraph();
getch();
return 0;
}
```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20C++%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32-bit windows 7, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>

Lab 2

Write the C program that will write draw the line using incremental line algorithm starting at the pixel position from 25th row and 30th column.

```
#include <stdio.h>
#include <conio.h>
#include <graphics.h>
#include <math.h>
```

```
void lineIncremental(int x1, int y1, int x2, int y2){
```

```

int x;
float dy = y2-y1;
float dx = x2-x1;
float m = dy/dx;
float y = y1;
if ((m > -1) || (m < 1) ){
    for(x=x1; x<=x2;x++){
        putpixel(x, floor(0.5+y), WHITE);
        y+=m;
    }
}
else{
    for(x=y1; x<=y2;x++){
        putpixel(x, floor(0.5+y), WHITE);
        x+=1/m;
    }
}
}

int main(){
    int gd=DETECT, gm;
    initgraph(&gd, &gm, "");
    lineIncremental(25, 30, 300, 400);
    lineIncremental(100, 400, 300, 10);
    getch();
    return 0;
}

```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20CPP%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32 windows, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>

Lab 3

Write the C program for drawing line by using Digital Differential Analyzer (DDA) Algorithm, straight line start on the screen starting at pixel position 30th row and 30th column.

```
#include<stdio.h>
#include<graphics.h>

//Function for finding absolute value
int abs (int n)
{
    return ( (n>0) ? n : ( n * (-1)));
}

//DDA Function for line generation
void DDA(int X0, int Y0, int X1, int Y1)
{
    // calculate dx & dy
    int dx = X1 - X0;
    int dy = Y1 - Y0;

    // calculate steps required for generating pixels
    int steps = abs(dx) > abs(dy) ? abs(dx) : abs(dy);

    // calculate increment in x & y for each steps
    float Xinc = dx / (float) steps;
    float Yinc = dy / (float) steps;

    // Put pixel for each step
    float X = X0;
    float Y = Y0;
```

```

for (int i = 0; i <= steps; i++)
{
    putpixel (X,Y,RED); // put pixel at (X,Y)
    X += Xinc;         // increment in x at each step
    Y += Yinc;         // increment in y at each step
    delay(100);        // for visualization of line-
                        // generation step by step
}
}

// Driver program
int main()
{
    int gd = DETECT, gm;

    // Initialize graphics function
    initgraph (&gd, &gm, "");

    int X0 = 2, Y0 = 2, X1 = 14, Y1 = 16;
    DDA(30, 30, 14, 16);
    getch();
    return 0;
}

```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20CPP%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32 windows, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>

Lab 4

Write the C Program for Bresenham's line drawing algorithm starting from specific pixel position.

```
#include<iostream.h>
#include<graphics.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 gdriver=DETECT, gmode, error, x0, y0, x1, y1;
```

```

initgraph(&gdriver, &gmode, "");
cout<<"Enter co-ordinates of first point: ";
cin>>x0>>y0;
cout<<"Enter co-ordinates of second point: ";
cin>>x1>>y1;
drawline(x0, y0, x1, y1);
getch();
return 0;
}

```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20CPP%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32 windows, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>

Lab 5

Write the C program to draw circle at the following pixel position at 45th row and 55th column by using Cartesian coordinate system.

```

#include<iostream.h>
#include<graphics.h>
#include<conio.h>
#include<math.h>
int main()

```



```

{
int xc, yc, x, y, r;
cout<<"please enter center point of the circle";
cin>>xc>>yc;
cout<<"please enter the radius of the circle";
cin>>r;
int gd=DETECT, gm;
initgraph(&gd,&gm, "");
for(x=r-xc; x<=r+xc; x++)
{
y=yc+sqrt((r*r-(x-xc)*(x-xc)));
putpixel(x,y, RED);
y=yc-sqrt((r*r-(x-xc)*(x-xc)));
putpixel(x,y, BLUE);
}
getch();
}

```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20CPP%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32 windows, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>

Lab 6

Write the C program to draw circle at the following pixel position at 55th row and 65th column by using polar coordinate system.

```
void CirclePolar(HDC hdc,int xc,int yc, int R)
{
    int x=R,y=0;
    double theta=0,dtheta=1.0/R;
    Draw8Points(hdc,xc,yc,x,y );
    while(x>y)
    {
        theta+=dtheta;
        x=round(R*cos(theta));
        y=round(R*sin(theta));
        Draw8Points(hdc,xc,yc,x,y);
    }
}
```

Mechanism to Conduct Lab:

Lab Session will be communicated through Skype / Adobe Connect session. As in computer graphics all the programs are implemented through Dev C++. During lab session students programming question are incorporated either through direct answers or through some sort of tutorials that are already uploaded at VULMS.

For 64-bit windows install the following version of Dev-CPP.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Dev-Cpp%205.9.2%20TDM-GCC%204.8.1%20Setup.rar>

For adding graphics library in 64-bit version of Dev-CPP see the link that illustrates the steps.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Adding%20graphics%20library%20in%20Dev%20C++%20For%20Windows%2010%20Tutorial.docx>

For 32-bit windows you can install the following version of Dev-CPP.

https://vulms.vu.edu.pk/Courses/CS602/Downloads/devcpp-4.9.9.2_setup.exe

In case you have 32 windows, you are required to follow the instruction as given in the following document.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/Add%20graphics%20in%20Dev%20cpp.doc>

Furthermore for 32-bit platform, you can download the graphics library from following link.

<https://vulms.vu.edu.pk/Courses/CS602/Downloads/graphics%20library.zip>