Name : Arun Mehta
Course- BCA (6th Sem)
Section- A
University Roll No- 1121023
Class Roll No - 21

Date : 16-06-2021
Subject : Computer Graphics and Animation

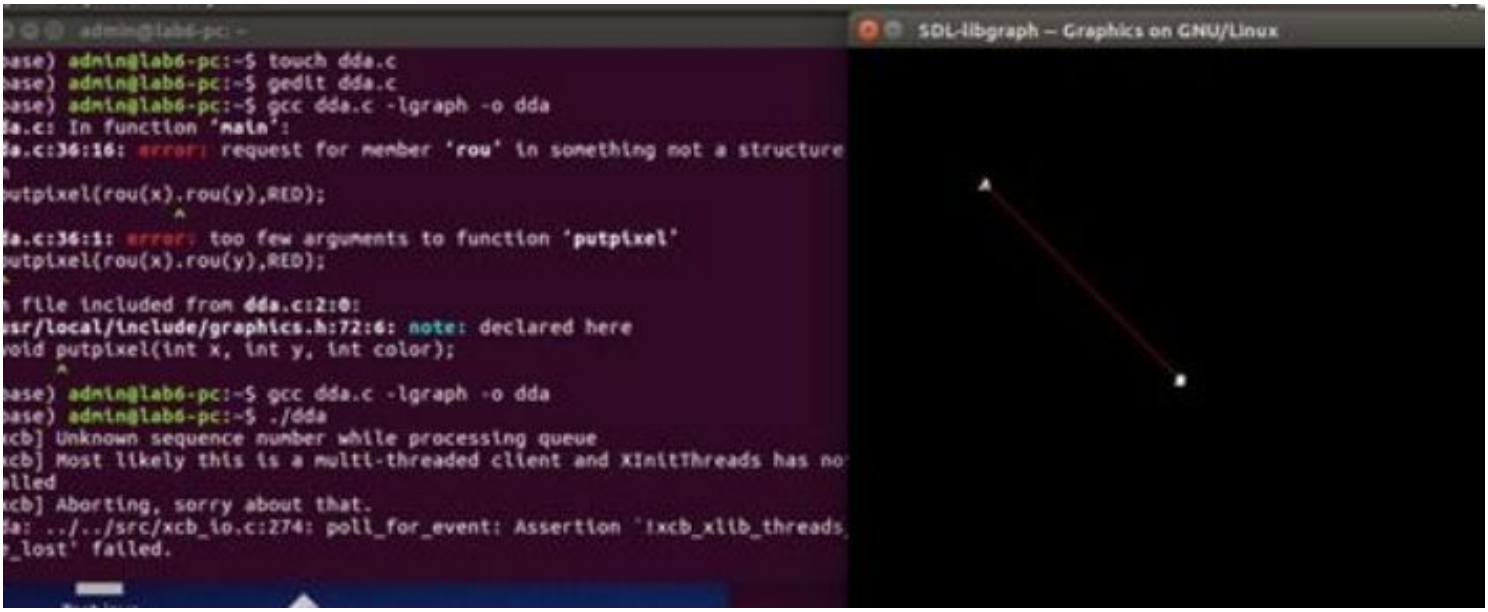
Ans- 1 Implementation of DDA Line Drawing Algorithm.

=> Algorithm :

Step 1 : Start
Step 2 : Declare $x_1, y_1, x_2, y_2, dx, dy, x, y$, as integer variable
Step 3 : Enter value of x_1, y_1, x_2, y_2
Step 4 : Calculate $dx = x_2 - x_1$
Step 5 : Calculate $dy = y_2 - y_1$
Step 6 : If ABS(dx) > ABS(dy)
Then step = abs(dx)
Else
Step 7 : $x_{inc} = dx/step$
$y_{inc} = dy/step$
assign $x = x_1$
assign $y = y_1$
Step 8 : set pixel (x, y)
Step 9 : $x = x + x_{inc}$
$y = y + y_{inc}$
Set pixels (Round (x), Round (y))
Step 10 : Repeat step 9 until $x = x_2$
Step 11 : End / Stop

=> Program:

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
void main()
{
    int gd= DETECT, gm, i;
    float x, y, dx, dy, steps;
    int x0, x1, y0, y1;
    initgraph(&gd, &gm, " ");
    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();
}
```


admin@lab6-pc: ~
base) admin@lab6-pc:~$ touch dda.c
base) admin@lab6-pc:~$ gedit dda.c
base) admin@lab6-pc:~$ gcc dda.c -lgraph -o dda
da.c: In function 'main':
da.c:36:16: error: request for member 'rou' in something not a structure
putpixel(rou(x).rou(y),RED);
da.c:36:1: error: too few arguments to function 'putpixel'
putpixel(rou(x).rou(y),RED);
file included from dda.c:2:0:
usr/local/include/graphics.h:72:6: note: declared here
void putpixel(int x, int y, int color);
base) admin@lab6-pc:~$ gcc dda.c -lgraph -o dda
base) admin@lab6-pc:~$ ./dda
xcb] Unknown sequence number while processing queue
xcb] Most likely this is a multi-threaded client and XInitThreads has no
alled
xcb] Aborting, sorry about that.
da: ../../src/xcb_io.c:274: poll_for_event: Assertion `!xcb_xlib_threads
_lost' failed.
SDL-libgraph – Graphics on GNU/Linux

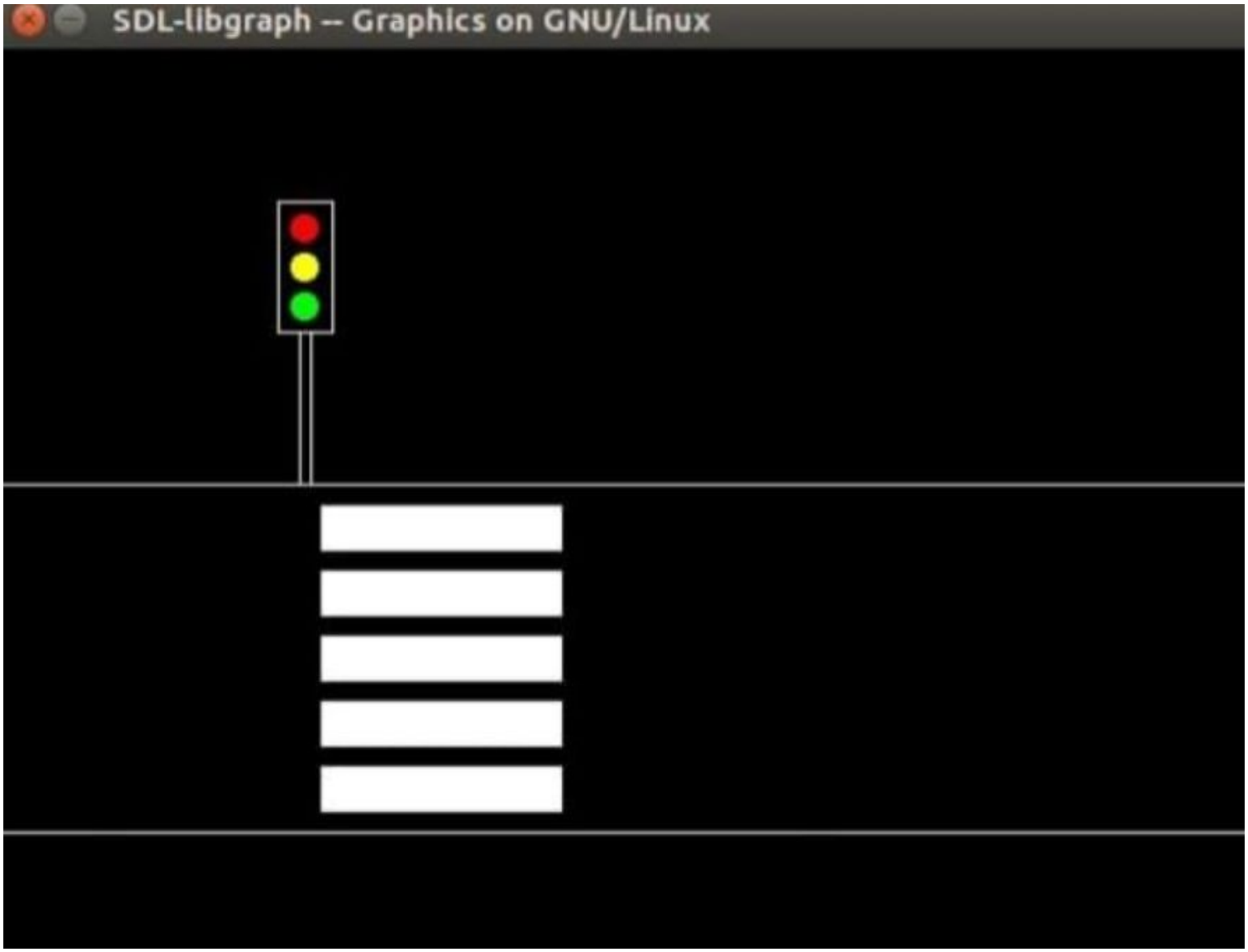
Name- Arun Mehta
Course- BCA (6th Sem)
Section- A
University Roll No- 1121023
Class Roll No- 21

Date - 16-06-2021
Subject - Computer Graphics And Animation

AN-3 Traffic Light Animation

```
#include<conio.h>
#include<graphics.h>
int main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"");
Line(0,200,getmaxx(),200);
line(0,360,getmaxx(),360);
setcolor(WHITE);
rectangle(150,210,260,230);
floodfill(152,220,WHITE);
rectangle(150,240,260,260)
floodfill(152,271,WHITE)
rectangle(150,270,260,290);
floodfill(152,271,WHITE);
rectangle(150,300,260,320);
floodfill(152,330,WHITE);
rectangle(150,330,260,350);
floodfill(152,331,WHITE);
```

```
setcolor(WHITE)
rectangle(140, 200, 145, 130);
setcolor(RED);
circle(142, 82, 6);
floodfill(142, 82, RED)
setcolor(YELLOW)
circle(142, 100, 6);
floodfill(142, 100, YELLOW)
setcolor(GREEN);
circle(142, 118, 6);
floodfill(143, 118, GREEN);
setcolor(WHITE);
rectangle(150, 180, 250, 300);
rectangle(250, 180, 420, 300);
rectangle(180, 250, 220, 300);
line(200, 100, 150, 180);
line(200, 100, 250, 180);
line(200, 100, 370, 100);
line(370, 100, 420, 180);
setcolor(BROWN);
floodfill(152, 182, WHITE);
floodfill(252, 182, WHITE);
setcolor(LIGHTRED);
floodfill(200, 105, WHITE);
floodfill(210, 105, WHITE);
getch();
closegraph();
return 0;
}
```


SDL-libgraph -- Graphics on GNU/Linux