

Name: Devashree Maharjan  
Course: BCA Sem: 6<sup>th</sup> Sec: A  
University Roll no: 1121041 (38)

Answer 1:

```
#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, "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;
```

Devashree

```
while (i <= steps)
{
    putpixel (x, y, RED);
    x += dx;
    y += dy;
    i = i + 1;
}
getch();
closegraph();
}
```

Devashree

Algorithm - DDA line drawing algorithm.

Step 1: Start Algorithm

Step 2: Declare  $x_1, y_1, x_2, y_2, dx, dy,$   
 $x, y$ , as integer variables.

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:  $xinc = dx / step$

$yinc = dy / step$

assign  $x = x_1$

assign  $y = y_1$

Step 8: setpixel ( $x, y$ )

Step 9:  $x = x + xinc$

$y = y + yinc$

setpixels (Round ( $x$ ), Round ( $y$ ))

Step 10: Repeat step 9 until  $x = x_2$

Step 11: End Algorithm.

Aakash



Name: Revashree Mahajan  
Course: BCA Sem: 6<sup>th</sup> Sec: A  
University Rollno: 1121041 (38)

Answer 3: Traffic light Animation -

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

```
int main()
```

```
{
```

```
    int gd = DETECT, gm;
```

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

```
    /* ROAD */
```

```
    line (0, 200, getmaxx(), 200);
```

```
    line (0, 360, getmaxx(), 360);
```

```
    /* Zebra Crossing */
```

```
    setcolor (WHITE);
```

```
    rectangle (150, 210, 260, 230);
```

```
    floodfill (152, 220, WHITE);
```

```
    rectangle (150, 240, 260, 260);
```

```
    floodfill (152, 241, WHITE);
```

```
    rectangle (150, 270, 260, 290);
```

```
    floodfill (152, 271, WHITE);
```

```
    rectangle (150, 300, 260, 320);
```

```
    floodfill (152, 301, WHITE);
```

```
    rectangle (150, 330, 260, 350);
```

```
    floodfill (152, 331, WHITE);
```



/\* Traffic Light \*/

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);

Devaashna

Hoodfill (200, 105, WHITE);

Hoodfill (210, 105, WHITE);

getch ();

closegraph ();

return 0;

}

Douglas

