

Name → GAUTAM PASWAN

Subject → computer graphics

Roll No → 1121053

Course → BCA

Solve 3 #include <stdio.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, 241, WHITE);



SetColor (WHITE);

rectangle (140, 200, 145, 130);

rectangle (130, 130, 155, 70);

SetColor (RED);

Circle (142, 82, 6);

Floodfill (142, 82, RED);

SetColor (YELLOW);

Circle (142, 60, 6);

Floodfill (142, 100, YELLOW);

SetColor (GREEN);

Circle (142, 118, 6);

Floodfill (143, 118, GREEN);

SetColor (WHITE);

getch();

closegraph();

return 0;

}



Solve 1

#include &lt;stdio.h&gt;

#include &lt;graphics.h&gt;

int main()

{

int xon(float num)

{

return num &lt; 0 ? num - 0.5 : num + 0.5

}

int x1 = 100, x2 = 250, y1 = 100, y2 = 250, step;

int gd = DETECT, gm;

float x, y, m;

int dx = x2 - x1;

int dy = y2 - y1;

m = dy / dx;

if (dx &gt; dy)

step = dx;

else

step = dy;



initGraph (sgd, sgm, " ");

outtent xy (x<sub>1</sub>, y<sub>1</sub>, "A");

outtent (x<sub>2</sub>, y<sub>2</sub>, "B");

PutPixel (x<sub>1</sub>, y<sub>1</sub>, RED);

x = x<sub>1</sub>, y = y<sub>1</sub>;

while (step > 0)

{

if (m < 1)

{

x = x + 1;

y = y + m;

}

if (m >= 1)

{

x = x + 1/m;

y = y + 1;

}

PutPixel (x<sub>0</sub>(x), x<sub>0</sub>(y), RED);

Step = 5

}

getch();

return 0;

}



# 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:- initialize the 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  $dx > dy$   
Then,  $Step = dx$   
else  $Step = dy$

Step 7 Assign,  $x = x_1$   
 $y = y_1$



Step 8 :- Set Pixel  $(x, y)$

Step 9 :- Starting while loop (step 8)

if,  $m < 1$

then  $x = x + 1$ ;

$y = y + m$ ;

and,

if  $m > 1$

then

$x = x + 1/m$ ;

$y = y + 1$ ;

Step 10 :- Set Pixel  $(\text{Rev}(x), \text{Rev}(y))$

Step 11 :- Repeat Step 9 until  $x = x_2$

Step 12 :- End Algorithm



