Program 3: Write a program to implement Boundary-fill algorithm.

#include <graphics.h>

void boundaryFill4(int x, int y, int fill\_color,int boundary\_color)

{

if(getpixel(x, y) != boundary\_color &&

getpixel(x, y) != fill\_color)

{

putpixel(x, y, fill\_color);

boundaryFill4(x + 1, y, fill\_color, boundary\_color);

boundaryFill4(x, y + 1, fill\_color, boundary\_color);

boundaryFill4(x - 1, y, fill\_color, boundary\_color);

boundaryFill4(x, y - 1, fill\_color, boundary\_color);

}

}

int main()

{

int gd = DETECT, gm;

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

int x = 250, y = 200, radius = 50;

circle(x, y, radius);

boundaryFill4(x, y, 6, 15);

delay(10000);

getch();

closegraph();

return 0;

Output:

