Name: Shivani

Rollno.:1022762(52)

Subject: computer graphics

Q1.Write a program to rotate an object by 90 degree in clockwise direction

#include<graphics.h>

#include<stdio.h>

#include<conio.h>

#include<math.h>

int main()

{

int gd=DETECT,gm;

int pivot\_x,pivot\_y,x,y;

double degree,radian;

int rotated\_point\_x,rotated\_point\_y;

initgraph(&gd,&gm,"C://TURBOC3//BGI");

cleardevice();

printf("\n Enter an initial coordinates of the line = ");

scanf("%d %d",&pivot\_x,&pivot\_y);

printf("\n Enter a final coordinates of the line = ");

scanf("%d %d",&x,&y);

line(pivot\_x,pivot\_y,x,y);

printf("\n\n Now, Enter a degree = ");

scanf("%lf",&degree);

radian=degree\*0.01745;

rotated\_point\_x=(int)(pivot\_x +((x-pivot\_x)\*cos(radian)-(y-pivot\_y)\*sin(radian)));

rotated\_point\_y=(int)(pivot\_y +((x-pivot\_x)\*sin(radian)+(y-pivot\_y)\*cos(radian)));

setcolor(RED);

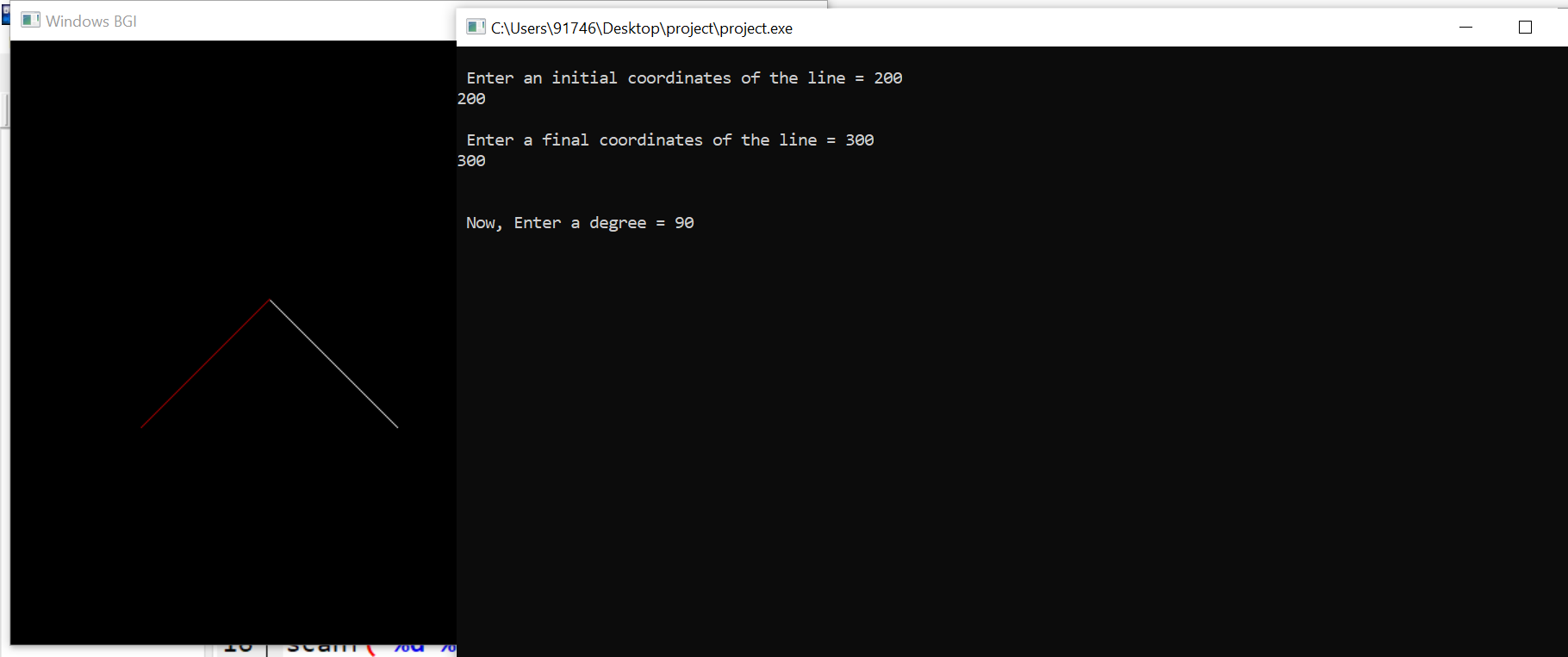
line(pivot\_x,pivot\_y,rotated\_point\_x,rotated\_point\_y);

getch();

closegraph();

}

Output:



Name: Shivani

Rollno.:1022762(52)

Subject: computer graphics

Q2. Write a program to draw a line using Bresenham’s line generation algorithm.

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

int main()

{

int x,y,x1,y1,x2,y2,p,dx,dy;

int gd=DETECT,gm;

initgraph(&gd,&gm, NULL);

printf("\nEnter the x-coordinate of the first point ::");

scanf("%d",&x1);

printf("\nEnter the y-coordinate of the first point ::");

scanf("%d",&y1);

printf("\nEnter the x-coordinate of the second point ::");

scanf("%d",&x2);

printf("\nEnter the y-coordinate of the second point ::");

scanf("%d",&y2);

x=x1;

y=y1;

dx=x2-x1;

dy=y2-y1;

putpixel(x,y,2);

p=(2\*dy-dx);

while(x<=x2)

{

if(p<0)

{

x=x+1;

p=p+2\*dy;

}

else

{

x=x+1;

y=y+1;

p=p+(2\*dy)-(2\*dx);

}

putpixel(x,y,7);

}

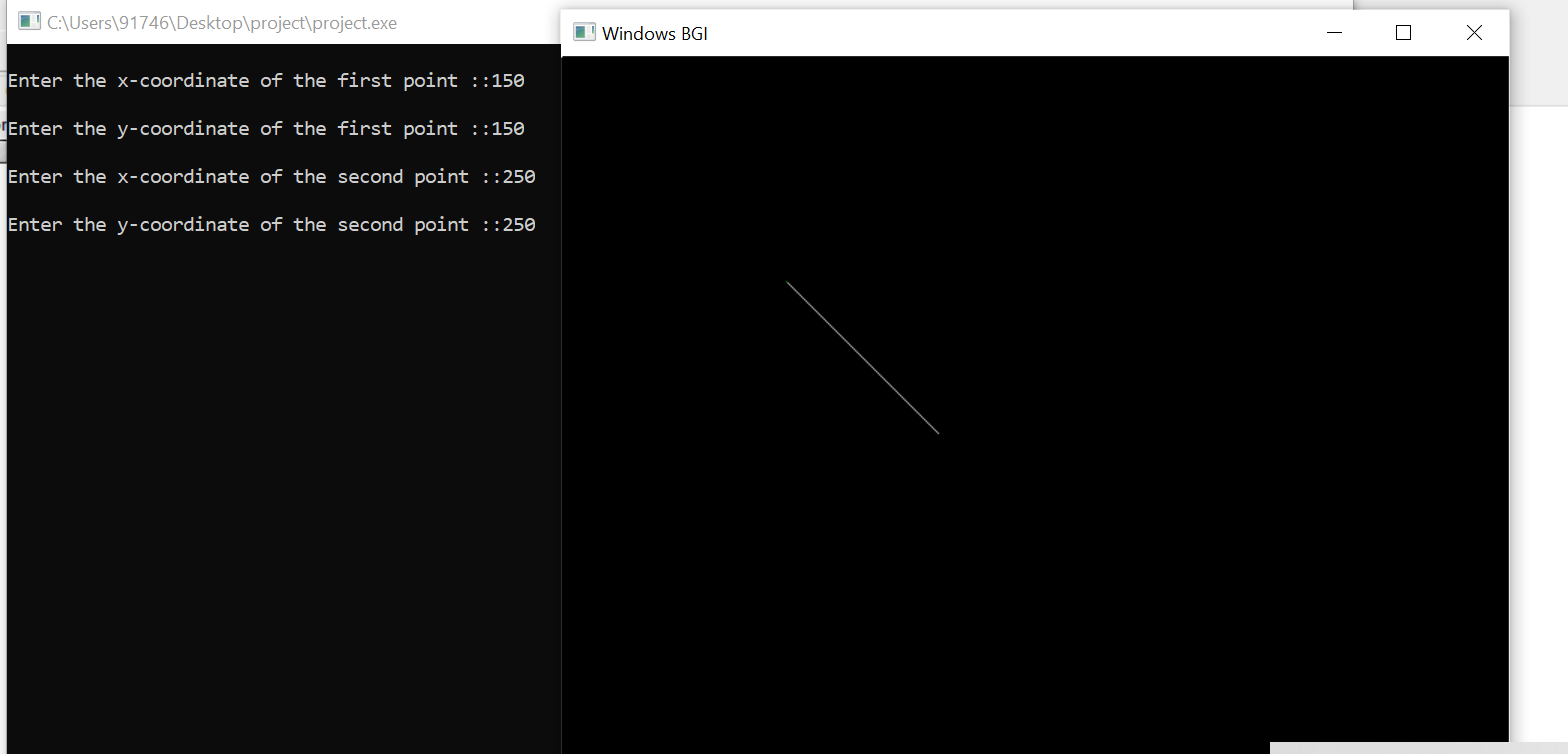
getch();

closegraph();

return 0;

}

Output:



Name: Shivani

Rollno.:1022762(52)

Subject: computer graphics

Q3. Write a program to implement Boundary-fill algorithm.

#include<stdio.h>

#include<graphics.h>

void boundary\_fill(int x,int y,int boundary\_color,int fill\_color);

int main()

{

int gd,gm,x,y,x1,x2,y1,y2;

detectgraph(&gd,&gm);

initgraph(&gd,&gm,"C://TurboC3//BGI");

printf("Enter top-left point of rectangle: ");

scanf("%d%d",&x1,&y1);

printf("Enter bottom-right point of rectangle: ");

scanf("%d%d",&x2,&y2);

cleardevice();

setcolor(WHITE);

rectangle(x1,y1,x2,y2);

boundary\_fill(x1+1,y1+1,15,YELLOW);

getch();

closegraph();

return 0;

}

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

{

int current;

current=getpixel(x,y);

if(current!=boundary\_color && current!=fill\_color)

{

putpixel(x,y,fill\_color);

delay(20);

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

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

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

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

}

}

Output:

