

Lab Report-02

Course Title: Computer Graphics Laboratory

Course Code: CSE-304

3rd Year 1st Semester Examination 2022

Date of Submission: 04.06.2023



Submitted to-

Dr. Mohammad Shorif Uddin

Professor

Dr. Morium Akter

Associate Professor

*Department of Computer Science & Engineering
Jahangirnagar University*

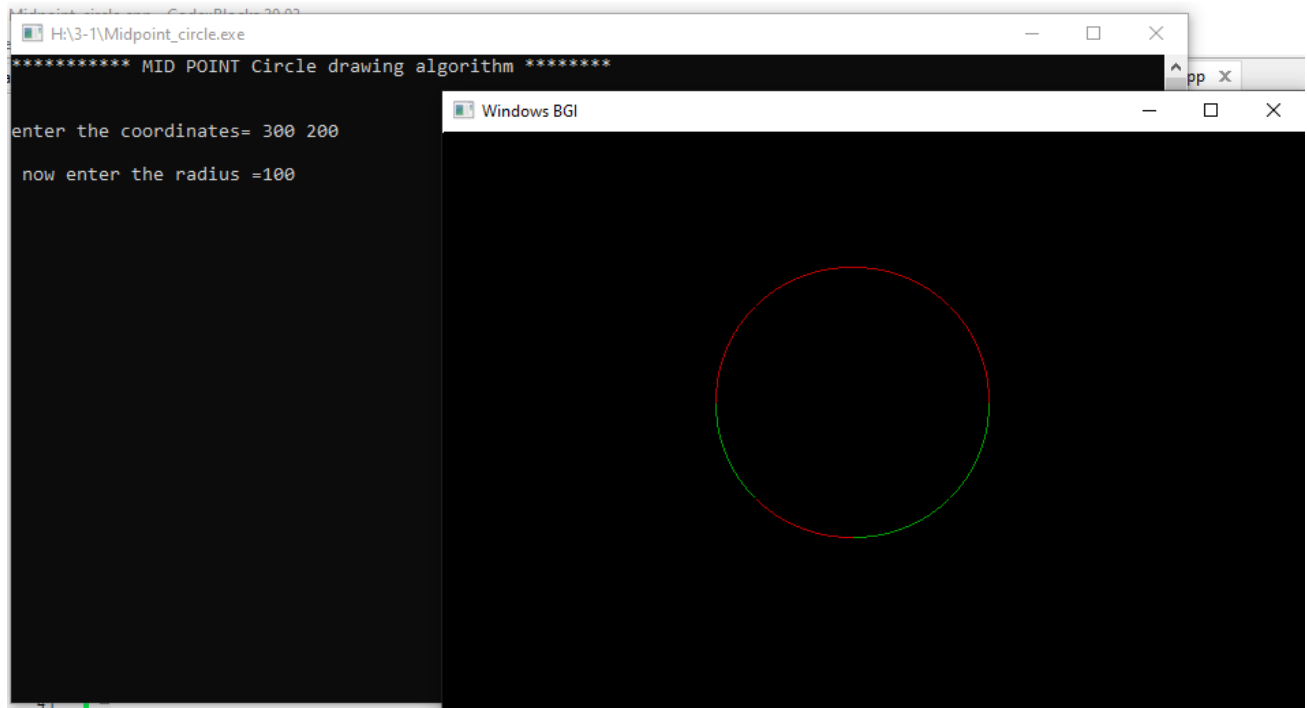
Sl	Class Roll	Exam Roll	Name
01	380	202192	Sovon Mallick

Scan Conversion of a Circle using Midpoint Algorithm

Source Code:

<pre>#include<conio.h> #include<stdio.h> int main() { int x,y,x_mid,y_mid,radius,dp; int g_mode,g_driver=DETECT; initgraph(&g_driver,&g_mode,"C:\\\\TURBOC3\\\\BGI"); printf("***** MID POINT Circle drawing algorithm *****\n\n"); printf("\nenter the coordinates= "); scanf("%d %d",&x_mid,&y_mid); printf("\n now enter the radius ="); scanf("%d",&radius); x=0; y=radius; dp=1-radius; do { putpixel(x_mid+x,y_mid+y,GREEN); putpixel(x_mid+y,y_mid+x,GREEN);</pre>	<pre>putpixel(x_mid-y,y_mid+x,GREEN); putpixel(x_mid-x,y_mid+y,RED); putpixel(x_mid-x,y_mid-y,RED); putpixel(x_mid-y,y_mid-x,RED); putpixel(x_mid+y,y_mid-x,RED); putpixel(x_mid+x,y_mid-y,RED); if(dp<0) { dp+=(2*x)+1; } else { y=y-1; dp+=(2*x)-(2*y)+1; } x=x+1; } while(y>x); getch(); }</pre>
--	---

Output:



Scan Conversion of an Ellipse

Source Code:

<pre> #include<stdio.h> #include<conio.h> #include<graphics.h> #include<math.h> void disp(); float x,y; int xc,yc; int main() { int gd=DETECT,gm,a,b; float p1,p2; initgraph(&gd,&gm,"c:\\turbo3\\bgi"); printf("*** Ellipse Generating Algorithm ***\n"); printf("Enter the value of Xc\t"); scanf("%d",&xc); printf("Enter the value of yc\t"); scanf("%d",&yc); printf("Enter X axis length\t"); scanf("%d",&a); printf("Enter Y axis length\t"); scanf("%d",&b); x=0;y=b; disp(); p1=(b*b)-(a*a*b)+(a*a)/4; while((2.0*b*b*x)<=(2.0*a*a*y)) { x++; if(p1<=0) p1=p1+(2.0*b*b*x)+(b*b); else { y--; p1=p1+(2.0*b*b*x)+(b*b)- (2.0*a*a*y); } disp(); } } </pre>	<pre> x=-x; disp(); x=-x; delay(50); } x=a; y=0; disp(); p2=(a*a)+2.0*(b*b*a)+(b*b)/4; while((2.0*b*b*x)>(2.0*a*a*y)) { y++; if(p2>0) p2=p2+(a*a)-(2.0*a*a*y); else { x--; p2=p2+(2.0*b*b*x)- (2.0*a*a*y)+(a*a); } disp(); y=-y; disp(); y=-y; delay(50); } getch(); closegraph(); } void disp() { putpixel(xc+x,yc+y,7); putpixel(xc-x,yc+y,7); putpixel(xc+x,yc-y,7); putpixel(xc-x,yc-y,7); } </pre>
--	---

Output:

