ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

**Maitighar, Kathmandu**

****

**Computer Graphics**

**Lab Assignment**

**SUBMITTED BY**

**Pratik Gautam**

**013BSCCSIT029**

**4th sem/ 2nd year**

**SUBMITTED TO**

**Er. Anil K Sah**

**Lecturer**

**Department of Computer Science**

**St. Xavier’s College**

**STATEMENT**

Draw a circle using midpoint algorithm in c++ programming.

**ALGORITHM**

1        Input centre (xc,yc) and rx and ry for the ellipse and obtain the first point as

(x0,y0)=(0,ry)

2.  Calculate initial decision parameter value in region 1 as

P10=ry2-rx2ry+1/4rx2

3.      At each xk position in region 1,starting at k=0,compute xk+1=xk+1

If p1k<0, then the next point to plot is

 p1k+1=p1k+2ry2xk+1+ry2

yk+1= yk

otherwise next point to plot is

 yk+1=yk-1

 p1k+1=p1k+2ry2xk+1+ry2-2rx2yk+1with xk+1=xk+1 and yk+1=yk-1

4.      Calculate the initial value- of decision parameter at region 2 using last calculated point say (x0,y0) in region 1 as

P20=ry2(x0+1/2)2+rx2(y0-1)2-rx2y2

5.      At each yk position in Region 2 starting at k=0, perform computation

Yk+1=y-1;

If p2k>0,then

Xk+1=xk

P2k+1=p2k-2rx2(yk-1)+rx2

Otherwise

Xk+1=xk+1

P2k+1=p2k+2ry2xk+1-2rx2yk+1+rx2

where xk+1=xk+1 and yk+1=yk-1

6.      Determine the symmetry points in other  3 quadrants.

7.       Move each calculated point (xk, yk) on to the centered (xc,yc) ellipse path as

xk = xk + xc;

yk = yk + yc

8.      Repeat the process for region 1 until and region until 2ry2xk>=2rx2yk and region until (xk, yk) =(rx,0)

**SOURCE CODE**

//---------------------------------------------------------------------------

#include <vcl\vcl.h>

#pragma hdrstop

#include "Unit1.h"

//---------------------------------------------------------------------------

#pragma resource "\*.dfm"

TForm1 \*Form1;

int xc,yc,rx,ry,x,y,p;

//---------------------------------------------------------------------------

\_\_fastcall TForm1::TForm1(TComponent\* Owner)

: TForm(Owner)

{

}

//---------------------------------------------------------------------------

void \_\_fastcall TForm1::Button1Click(TObject \*Sender)

{

xc=StrToInt(Edit1->Text);

yc=StrToInt(Edit2->Text);

rx=StrToInt(Edit3->Text);

ry=StrToInt(Edit4->Text);

x=0;

y=ry;

p=(ry\*ry)-(rx\*rx\*ry)+((rx\*rx)/4);

while((2\*x\*ry\*ry)<(2\*y\*rx\*rx))

{

Image1->Canvas->Pixels[xc+x][yc-y]=RGB(100,125,150);

Image1->Canvas->Pixels[xc-x][yc+y]=RGB(100,1,0);

Image1->Canvas->Pixels[xc+x][yc+y]=RGB(0,125,150);

Image1->Canvas->Pixels[xc-x][yc-y]=RGB(200,15,150);

if(p<0)

{

x=x+1;

p=p+(2\*ry\*ry\*x)+(ry\*ry);

}

else

{

x=x+1;

y=y-1;

p=p+(2\*ry\*ry\*x+ry\*ry)-(2\*rx\*rx\*y);

}

}

p=((float)x+0.5)\*((float)x+0.5)\*ry\*ry+(y-1)\*(y-1)\*rx\*rx-rx\*rx\*ry\*ry;

while(y>=0)

{

Image1->Canvas->Pixels[xc+x][yc-y]=RGB(100,125,150);

Image1->Canvas->Pixels[xc-x][yc+y]=RGB(100,1,0);

Image1->Canvas->Pixels[xc+x][yc+y]=RGB(0,125,150);

Image1->Canvas->Pixels[xc-x][yc-y]=RGB(200,15,150);

if(p>0)

{

y=y-1;

p=p-(2\*rx\*rx\*y)+(rx\*rx);

}

else

{

y=y-1;

x=x+1;

p=p+(2\*ry\*ry\*x)-(2\*rx\*rx\*y)-(rx\*rx);

}

}

**OUTPUT SCREEN**

