ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Computer Graphics**

**Lab Assignment #6**

**Submitted by:**

PRANESH DHUNJU SHRESTHA

013BSCCSIT026

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Anil Sah**  **Lecturer,**  **St. Xavier’s College** |  |

**Submission Date**

**August 25, 2015**

**STATEMENT :**

DRAW A ELLIPSE USING MID-POINT ALGORITHM.

**MID POINT ELLIPSE ALGORITHM:**

**Step 1:**  Input x-axis radius *rx*, y-axis radius *ry*and ellipse center (Xc, Yc) from the user.  
  
**Step 2:**  Plot the initial set of points.  
             (Xc, Yc + ry)  
             (Xc, Yc - ry)  
  
**Step 3:**   Find the initial value of the decision parameter in region 1.  
           P10 = (ry^2)  - (rx^2)ry  + (1/4)rx^2    
  
**Step 4:**   If P1k < 0, the subsequent point along the ellipse centered at (0, 0) is  
              Xk + 1 = Xk + 1  
             Yk + 1 = Yk

                      and P1k + 1 = P1k + 2(ry^2)Xk+1  + ry^2

             Otherwise, the subsequent point along the ellipse is

              Xk + 1 = Xk + 1

              Yk + 1 = Yk - 1  
              and P1k + 1 = P1k + 2(ry^2)Xk+1 - 2(rx^2)Yk+1  + ry^2

              where 2(ry^2)Xk+1 = 2(ry^2)Xk + 2(ry^2)   and

                       2(rx^2)Yk+1 = 2(rx^2)Yk +2(rx^2)

**Step 5:** Find the initial value of the decision parameter in region 2.

           P20 = (ry^2)(X0 + 1/2)^2  - (rx^2)(y0 - 1)^2  + (rx^2)(ry^2)  
  
**Step 6:** If P2k > 0, the subsequent point along the ellipse centered at (0, 0) is  
             Xk + 1 = Xk  
             Yk + 1 = Yk - 1

                      and P2k + 1 = P2k - 2(rx^2)Yk+1  + rx^2

             Otherwise, the subsequent point along the ellipse is

              Xk + 1 = Xk + 1

              Yk + 1 = Yk - 1  
              and P2k + 1 = P2k + 2(ry^2)Xk+1 - 2(rx^2)Yk+1  + rx^2

              where 2(ry^2)Xk+1 = 2(ry^2)Xk + 2(ry^2)   and

                       2(rx^2)Yk+1 = 2(rx^2)Yk +2(rx^2)

**Step 7:**  Find the symmetric points in all other quadrants.

**Step 8:**  Plot the above calculated pixel(x, y) onto the elliptical path of an ellipse centered at (Xc, Yc) and plot the co-ordinates.

             (Xc + x, Yc + y)

             (Xc - x, Yc + y)  
             (Xc + x, Yc - y)  
             (Xc - x, Yc - y)

**Step 9:**  Repeat the above steps for region 1 until 2(ry^2)X >= 2(rx^2)Y

**Step 10:**  Repeat the above steps for region 2 until Y > 0.

**SOURCE CODE :**

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

#include <vcl\vcl.h>

#pragma hdrstop

#include "code.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(255,0,0);

Image1->Canvas->Pixels[xc-x][yc+y]=RGB(40,168,26);

Image1->Canvas->Pixels[xc+x][yc+y]=RGB(253,11,245);

Image1->Canvas->Pixels[xc-x][yc-y]=RGB(11,11,253);

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(255,0,0);

Image1->Canvas->Pixels[xc-x][yc+y]=RGB(40,168,26);

Image1->Canvas->Pixels[xc+x][yc+y]=RGB(253,11,245);

Image1->Canvas->Pixels[xc-x][yc-y]=RGB(11,11,253);

if(p>0)

{

y=y-1;

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

}

else

{

y=y-1;

x=x+1;

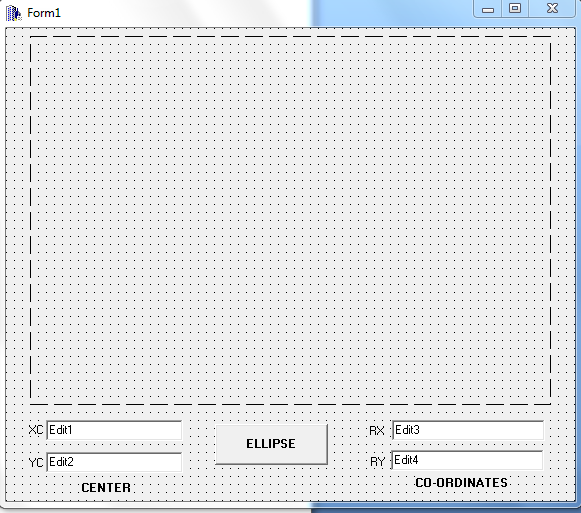
p=p+(2\*ry\*ry\*x)-(2\*rx\*rx\*y)-(rx\*rx);

}

}

}

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



**OUTPUT SCREEN :**

