**ST.XAVIER,S COLLEGE**

**Maitighar, Kathmandu**



Digital Logic Lab Assignment #5

**Draw a circle using midpoint circle algorithm.**

Yub Raj Basnet

013BScCSIT048 (4th Semester)

**Submitted to**

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

**Submitted date: August 21, 2015**

# Algorithm:

1. Input radius r and circle centre ( xc,yc )and obtain the first point on circle centered at origin as.

(x0,y0)=(0,r)

2. Calculate initial decision parameter

P0

3. At each xkposition, starting at *k* 0, performs the tests:

If pk0next point along the circle centre at (0,0) is (xk+1,yK )

Pk+1 =pk+2xk+1+1

Otherwise, the next point along circle is (xk+1, yK-1)

Pk+1=pk+2xk+1+1-2yk+1

Where 2xk+1=2xk+2 and 2yk+1=2yk-2

4. Determine symmetry point on the other seven octants.

5. Move each calculated pixels positions (*x*, *y*) in to circle path centered at ( xc,yc ) as

*x* *x* *xc* , *y* *y* *yc*

6. Repeat 3 through 5 until *x* *y*.

**Source Code**

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

#include <vcl\vcl.h>

#pragma hdrstop

#include "Unit1.h"

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

#pragma resource "\*.dfm"

TForm1 \*Form1;

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

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

: TForm(Owner)

{

}

void drawpoints(int x,int y, int xc,int yc)

{

Form1->Image1->Canvas->Pixels[xc+x][yc+y]=RGB(255,0,0);

Form1->Image1->Canvas->Pixels[xc-x][yc+y]=RGB(0,255,0);

Form1->Image1->Canvas->Pixels[xc+x][yc-y]=RGB(0,0,255);

Form1->Image1->Canvas->Pixels[xc-x][yc-y]=RGB(255,255,0);

Form1->Image1->Canvas->Pixels[xc+y][yc+x]=RGB(0,255,255);

Form1->Image1->Canvas->Pixels[xc-y][yc+x]=RGB(255,0,255);

Form1->Image1->Canvas->Pixels[xc+y][yc-x]=RGB(100,100,100);

Form1->Image1->Canvas->Pixels[xc-y][yc-x]=RGB(200,50,50);

}

void drawcircle(int xc,int yc,int r)

{

int p,x,y;

x=0;

y=r;

drawpoints(x,y,xc,yc);

p=1-r;

while(x<y)

{

if(p<0)

{

x=x+1;

p=p+2\*x+1;

}

else

{

x=x+1;

y=y-1;

p=p+2\*(x-y)+1;

}

drawpoints(x,y,xc,yc);

}

}

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

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

{

int xc,yc,r;

xc=StrToInt(Edit1->Text);

yc=StrToInt(Edit2->Text);

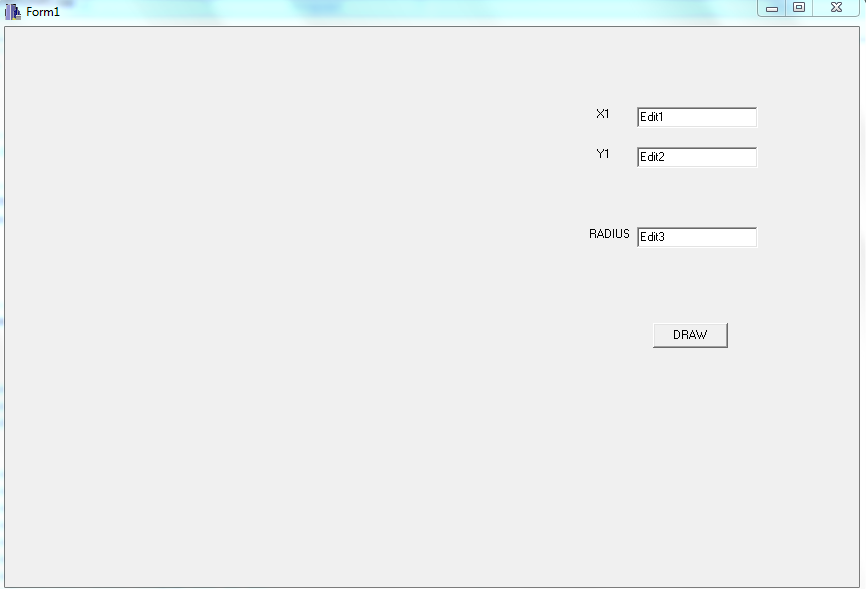
r=StrToInt(Edit3->Text);

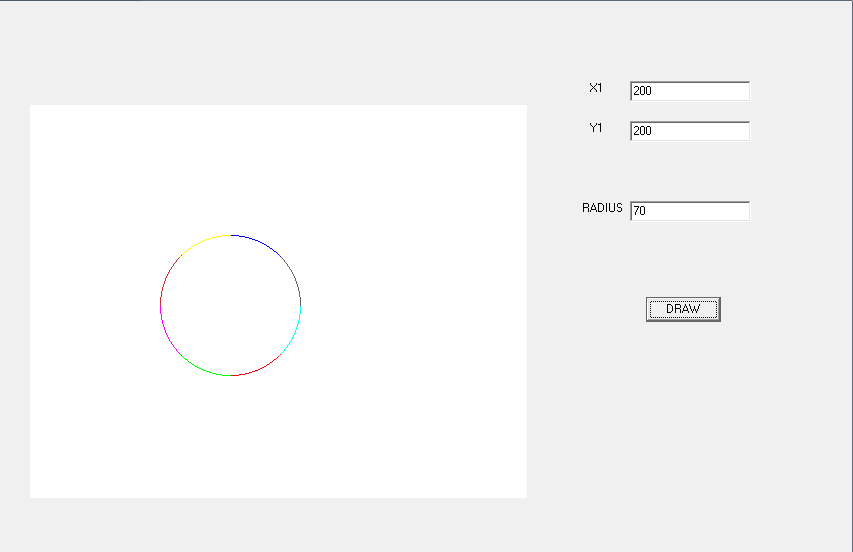
drawcircle(xc,yc,r);

}

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

# Output





# Conclusion:

Hence, we constructed the circle using midpoint algorithm using C++ builder.

# Reference

[1] Er.Anil Sah,”CG Manual( 2015 ).pdf”