ST. XAVIER’S COLLEGE

**Maitighar,Kathmandu**

**(Affiliated to Tribhuvan University)**



**Computer Graphics**

**Lab Assignment #3**

**Submitted By**

Arun Sanjel

B.Sc. CSIT

Year II/IV Semester

013BSCIT010

**Submitted To**

Er. Anil K. Sah

Lecturer

Department of Computer Science

St. Xavier’s College

Maitighar, Kathmandu

**Submitted On**

August 14, 2015

**STATEMENT**

DRAW A LINE USING DDA ALGORITHM WHERE USER PROVIDES THE LINE COORDINATES.

**SOURCE CODE**

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

#include <vcl\vcl.h>

#pragma hdrstop

#include<math.h>

#include "line.h"

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

#pragma resource "\*.dfm"

TForm1 \*Form1;

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

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

: TForm(Owner)

{

}

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

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

{

outPanel->Canvas->FillRect(ClientRect);

int delX,delY,inc,i;

float incX,incY,x,y;

int x1,x2,y1,y2;

int r,g,b;

x1 = StrToInt(inXone->Text);

x2 = StrToInt(inXtwo->Text);

y1 = StrToInt(inYone->Text);

y2 = StrToInt(inYtwo->Text);

r = StrToInt(inRed->Text);

g = StrToInt(inGreen->Text);

b = StrToInt(inBlue->Text);

delX = abs(x2 - x1);

delY = abs(y2 - y1);

if(delX>delY){

inc = delX;

}

else{

inc=delY;

}

incX = delX/inc;

incY = delY/inc;

//START Pixel

x=x1;

y=y1;

outPanel->Canvas->Pixels[x][y] = RGB (r,g,b);

for(i=1;i<=inc;i++){

x=x+incX;

y=y+incY;

outPanel->Canvas->Pixels[x][y] = RGB (r,g,b);

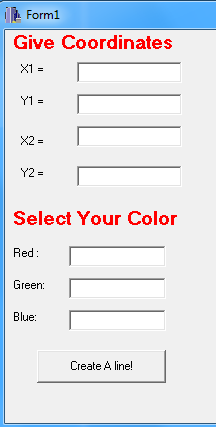
}

}

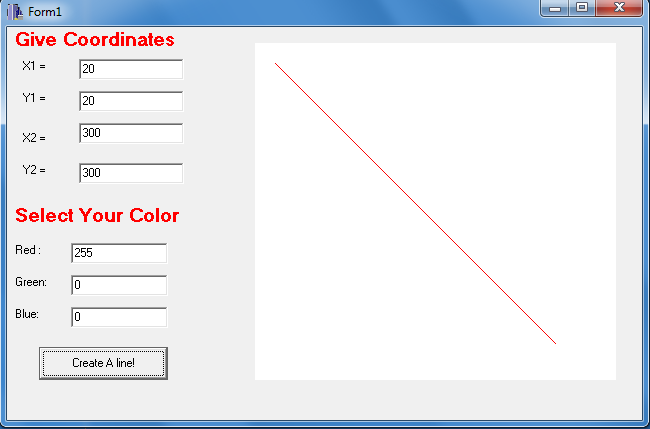
//---------------------------------------------------------------------------

**OUTPUT**

**INPUT FORM**

****

**OUTPUT PANEL**

****

**CONCLUSION**

Thus, using C++ builder DDA algorithm was implemented and result was displayed above.