**ST. XAVIER’S COLLEGE**

**(Affiliated to Tribhuvan University)**

**Maitighar, Kathmandu**

****

**COMPUTER GRAPHICS**

**LAB ASSIGNMENT#4**

**Submitted by:**

Shreesha Pokharel

013BSCCSIT038

**Submitted to:**

|  |  |
| --- | --- |
| **Er. Anil K. Sah** |  |

Lecturer

Department of Computer Science

Date of submission: 18th August, 2015

**Statement: Implementation of Bresenham’s Line Algorithm.**

Source code:

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

#include <vcl\vcl.h>

#pragma hdrstop

#include "Unit1.h"

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

#pragma resource "\*.dfm"

TForm1 \*Form1;

float x, y, dx, dy,Pk,n,xn,x1,y1,x2,y2;

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

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

: TForm(Owner)

{

}

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

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

{

x1=StrToInt(Edit1->Text);

y1=StrToInt(Edit2->Text);

x2=StrToInt(Edit3->Text);

y2=StrToInt(Edit4->Text);

dx=abs(x2-x1);

dy=abs(y2-y1);

if(x1>x2)

{

x=x2;

y=y2;

xn=x1;

}

else

{

x=x1;

y=y1;

xn=x2;

}

Image1->Canvas->Pixels[x][y] = RGB(180,150,100);

Pk = 2\*dy-dx;

while(x <= xn)

{

if(Pk<0)

{

x++;

Pk = Pk+2\*dy;

}

else

{

x++;

y++;

Pk=Pk-(2\*dy)+(2\*dx);

}

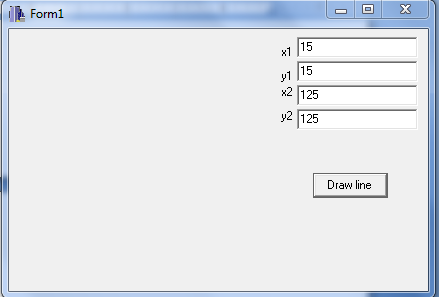
Image1->Canvas->Pixels[x][y] = RGB(180,150,100);

}

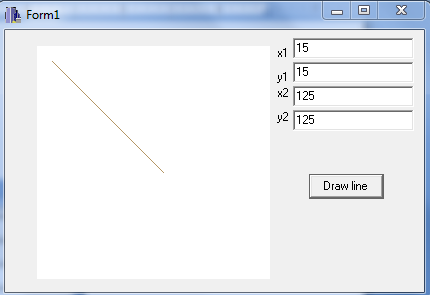
}

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

**Input:**

****

**Output:**

****

**Conclusion:**

Therefore, the implementation of Bresenham’s line algorithm was done and an output as a line of any desired color could be shown.

**Reference:**

[1] D. Hearn and M. Baker, Computer Graphics, second edition.