ST. XAVIER’S COLLEGE

**Maitighar, Kathmandu**

****

**Computer Graphics**

**Lab Assignment #10**

**SUBMITTED BY:**

**Siddhant Rimal**

**013BSCCSIT039**

**SUBMITTED TO**

|  |  |
| --- | --- |
| **Er. Anil Sah**  **( Lecturer )** |  |
| **Department of Computer Science** | |

Submission Date: 8th September 2015

**OBJECTIVE 10.1: TO CLIP A LINE**

|  |
| --- |
| **SOURCE CODE:** |
| //--------------------------------------------------------------------------- |
| #include <vcl\vcl.h> |
| #pragma hdrstop |
|  |
| #include "Line\_Clip.h" |
| //--------------------------------------------------------------------------- |
| #pragma resource "\*.dfm" |
| TForm1 \*Form1; |
| //-- Declaring UD Functions |
| void croppingboundaries(int X,int Y); |
| void clearall(void); |
| void flushvalues(void); |
| void reset(void); |
| //-- |
| //--DECLARING INTEGERS |
| int dx,dy,pk,nc=0,n; |
| int Xval,Yval,XMIN,YMIN,XMAX,YMAX,X1=0,Y1=0,X2=0,Y2=0,i,j; |
| int insiderun=0, insiderise=0; |
| int rightout=0, leftout=0; |
| int aboveout=0, belowout=0; |
| //-- |
|  |
| //--------------------------------------------------------------------------- |
| \_\_fastcall TForm1::TForm1(TComponent\* Owner) |
| : TForm(Owner) |
| { |
| xmin->Text=50; |
| ymin->Text=50; |
| xmax->Text=100; |
| ymax->Text=100; |
| lx1->Text=20; |
| ly1->Text=20; |
| lx2->Text=75; |
| ly2->Text=75; |
| } |
| //--------------------------------------------------------------------------- |
| void \_\_fastcall TForm1::CrtViewportClick(TObject \*Sender) |
| { |
| XMAX=StrToInt(xmax->Text); |
| XMIN=StrToInt(xmin->Text); |
| YMAX=StrToInt(ymax->Text); |
| YMIN=StrToInt(ymin->Text); |
| //Making a Viewport |
| Image1->Canvas->MoveTo(XMIN,YMIN); |
| Image1->Canvas->LineTo(XMIN,YMAX); |
| Image1->Canvas->MoveTo(XMIN,YMIN); |
| Image1->Canvas->LineTo(XMAX,YMIN); |
| Image1->Canvas->MoveTo(XMAX,YMIN); |
| Image1->Canvas->LineTo(XMAX,YMAX); |
| Image1->Canvas->MoveTo(XMAX,YMAX); |
| Image1->Canvas->LineTo(XMIN,YMAX); |
| //Made a Viweport |
| } |
| //--------------------------------------------------------------------------- |
| void \_\_fastcall TForm1::ClipLineClick(TObject \*Sender) |
| { |
| clearall(); |
| X1=StrToInt(lx1->Text); |
| Y1=StrToInt(ly1->Text); |
| X2=StrToInt(lx2->Text); |
| Y2=StrToInt(ly2->Text); |
| XMAX=StrToInt(xmax->Text); |
| XMIN=StrToInt(xmin->Text); |
| YMAX=StrToInt(ymax->Text); |
| YMIN=StrToInt(ymin->Text); |
|  |
|  |
| //Making a Viewport |
| Image1->Canvas->MoveTo(XMIN,YMIN); |
| Image1->Canvas->LineTo(XMIN,YMAX); |
| Image1->Canvas->MoveTo(XMIN,YMIN); |
| Image1->Canvas->LineTo(XMAX,YMIN); |
| Image1->Canvas->MoveTo(XMAX,YMIN); |
| Image1->Canvas->LineTo(XMAX,YMAX); |
| Image1->Canvas->MoveTo(XMAX,YMAX); |
| Image1->Canvas->LineTo(XMIN,YMAX); |
| //Made a Viweport |
|  |
| //Performing BLA |
| dx=X2-X1; |
| dy=Y2-Y1; |
| n=dx; |
| pk=2\*dy-dx; |
| Xval=X1; |
| Yval=Y1; |
| do{ |
| //--->This part is conditional plotting |
| croppingboundaries(Xval,Yval); |
| if(insiderun==1 ||(insiderun==0 && rightout==0 && leftout==0 && aboveout==0 && belowout==0)) |
| { |
| reset(); |
| //Draw the point on Canvas |
| Image1->Canvas->Pixels[Xval][Yval]=RGB(0,0,255); |
| } |
| //--->conditional plotting end |
| if (pk<0) //BLA contd |
| { |
| Xval=Xval+1; |
| pk=pk+2\*dy; |
| } |
| else |
| { |
| Xval=Xval+1; |
| Yval=Yval+1; |
| pk=pk+2\*dy-2\*dx; |
| } |
| nc++; |
| }while(nc<n); //BLA End |
| } |
|  |
| void croppingboundaries(int X,int Y){ |
| if (XMIN<X && X<XMAX) |
| {insiderun=1;} |
| else |
| {insiderun=0; |
| if (XMIN>X && X>XMAX) |
| {rightout=1;} |
| else |
| {rightout=0;} |
| if (XMIN<X && X<XMAX) |
| {leftout=1;} |
| else |
| {leftout=0;} |
| } |
|  |
| if (Y>YMIN && Y<YMAX) |
| {insiderise=1;} |
| else |
| {insiderise=0; |
| if (Y<YMIN && Y<YMAX) |
| {aboveout=1;} |
| else |
| {aboveout=0;} |
| if (Y>YMIN && Y>YMAX) |
| {belowout=1;} |
| else |
| {belowout=0;} |
| } |
| } |
|  |
|  |
| //--------------------------------------------------------------------------- |
| void \_\_fastcall TForm1::Button4Click(TObject \*Sender) |
| { |
| X1=StrToInt(lx1->Text); |
| Y1=StrToInt(ly1->Text); |
| X2=StrToInt(lx2->Text); |
| Y2=StrToInt(ly2->Text); |
|  |
| //Performing BLA |
| dx=X2-X1; |
| dy=Y2-Y1; |
| n=dx; |
| pk=2\*dy-dx; |
| Xval=X1; |
| Yval=Y1; |
| do{ |
| Image1->Canvas->Pixels[Xval][Yval]=RGB(0,0,255); |
| if (pk<0) //BLA contd |
| { |
| Xval=Xval+1; |
| pk=pk+2\*dy; |
| } |
| else |
| { |
| Xval=Xval+1; |
| Yval=Yval+1; |
| pk=pk+2\*dy-2\*dx; |
| } |
| nc++; |
| }while(nc<n); //BLA End |
| flushvalues(); |
| } |
| //--------------------------------------------------------------------------- |
| //--------------------------------------------------------------------------- |
| void \_\_fastcall TForm1::Button3Click(TObject \*Sender) |
| { |
| clearall(); |
| flushvalues(); |
| } |
| //--------------------------------------------------------------------------- |
| void clearall(void){ |
| Form1->Image1->Canvas->Brush->Color=clWhite; |
| Form1->Image1->Canvas->Brush->Style=bsSolid; |
| Form1->Image1->Canvas->FillRect(Rect(0,0,Form1->Image1->Width,Form1->Image1->Height)); |
| } |
|  |
| void flushvalues(void){ |
| Xval=0;Yval=0;pk=0;n=0;nc=0;Xval=0;Yval=0; |
| } |
| void reset(void){ |
| insiderun=0;insiderise=0; |
| rightout=0;leftout=0; |
| aboveout=0; belowout=0; |
| } |

**OUTPUT:**

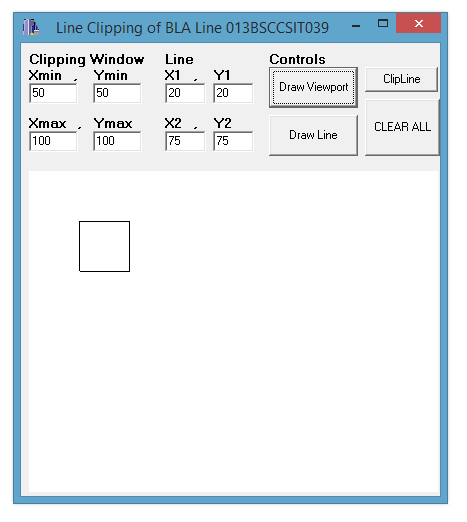


Fig: Drawing a Viewport

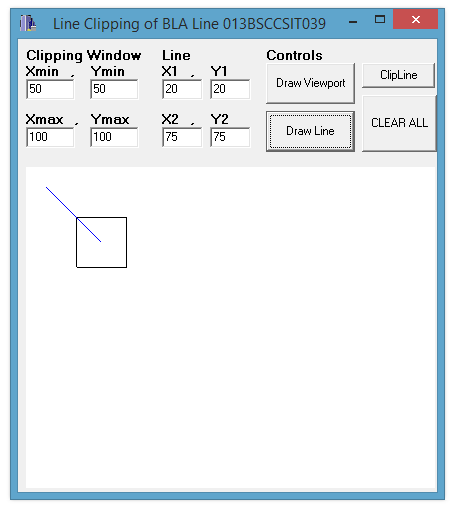


Fig: Drawing a Line

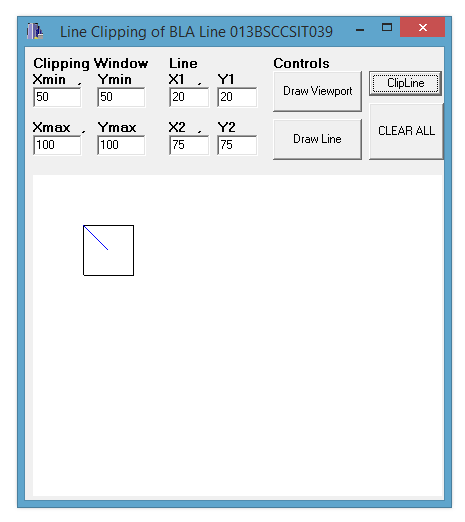


Fig: Clipping the drawn Line

**CONCLUSION:**

The program could define the parameters of the viewport such that when the line was drawn, if it would go beyond the boundary of the viewport, it would get clipped once the clipping algorithm was implemented.