Rajdhani
DATE / /

GROUP A Assignment 5 [CG]

Name : Aditya Onkar Patil
Batch : G4 (SE4)
Roll No: 21449
Performance Date : 27 / 10 / 2021
Submission Date : 28 / 10 / 2021

TITLE: Cohen Southerland Line Clipping Algorithm
PROBLEM STATEMENT: Write a C++ program to implement Cohen Southerland Line Clipping Algorithm.

LEARNING OBJECTIVES: To learn the concept of Cohen Southerland Line Clipping Algorithm as a concept of object oriented programming in computer graphics.

LEARNING OUTCOMES: After completion of this assignment students will be able to :
1. Implement the Cohen Southerland Line Clipping algorithm to clip the unnecessary part of figures outside the viewing window.

S/W AND H/W REQUIREMENTS: 64-bit open source Linux or its derivative, open source C++ programming tools like GCC / G++, Qt Creator, OpenGL.

REFERENCES: 1. Programming Principles and Practice using C++, Bjarne Stroustrup
2. www.qt.io.

Rajdhani
DATE / /

CONCEPT RELATED THEORY: COHEN SOUTHERLAND ALGORITHM:
The Cohen-Southerland algorithm is a computer graphics algorithm used for line clipping.
The Algorithm divides a 2-dimensional space into 9 spaces and then efficiently determines the lines and positions of lines that are visible in the central region of interest.

Step ①: Calculate the positions of both endpoints of the line and assign binary bits accordingly.

Step ②: Perform OR operation on both of these end points

Step ③: If OR operation gives 0000, then line is considered visible.
else: Perform AND operation on both end points:
if And ≠ 0000: then line is invisible
else: Line is clipped

Step ④: If a line is clipped case find its intersection with the boundaries of the window.

$m = (y_2 - y_1)/(x_2 - x_1)$

(a) If bit 1 is "1" line intersects left boundary
$y_3 = y_1 + m(x - x_1)$
where $x = x_{min}$

(b) If bit 2 is "1" line intersects right boundary
$y_3 = y_1 + m(x - x_1)$
where $x = x_{max}$

(c) If bit 3 is "1" line intersects bottom boundary
$x_3 = x_1 + (y - y_1)/m$
where $y = y_{min}$

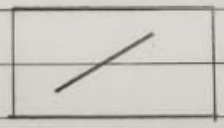
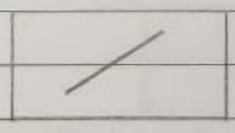
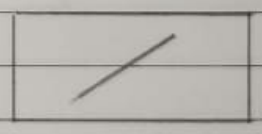
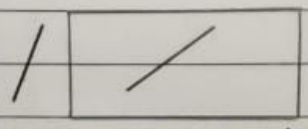
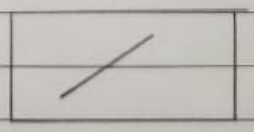
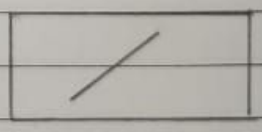
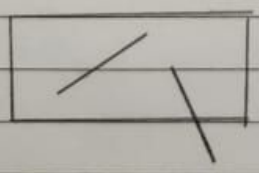
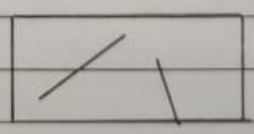
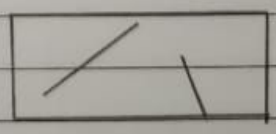
(d) If bit 4 is "1" line intersects top boundary
$x_3 = x_1 + (y - y_1)/m$ where $y = y_{max}$

Rajdhani
DATE / /

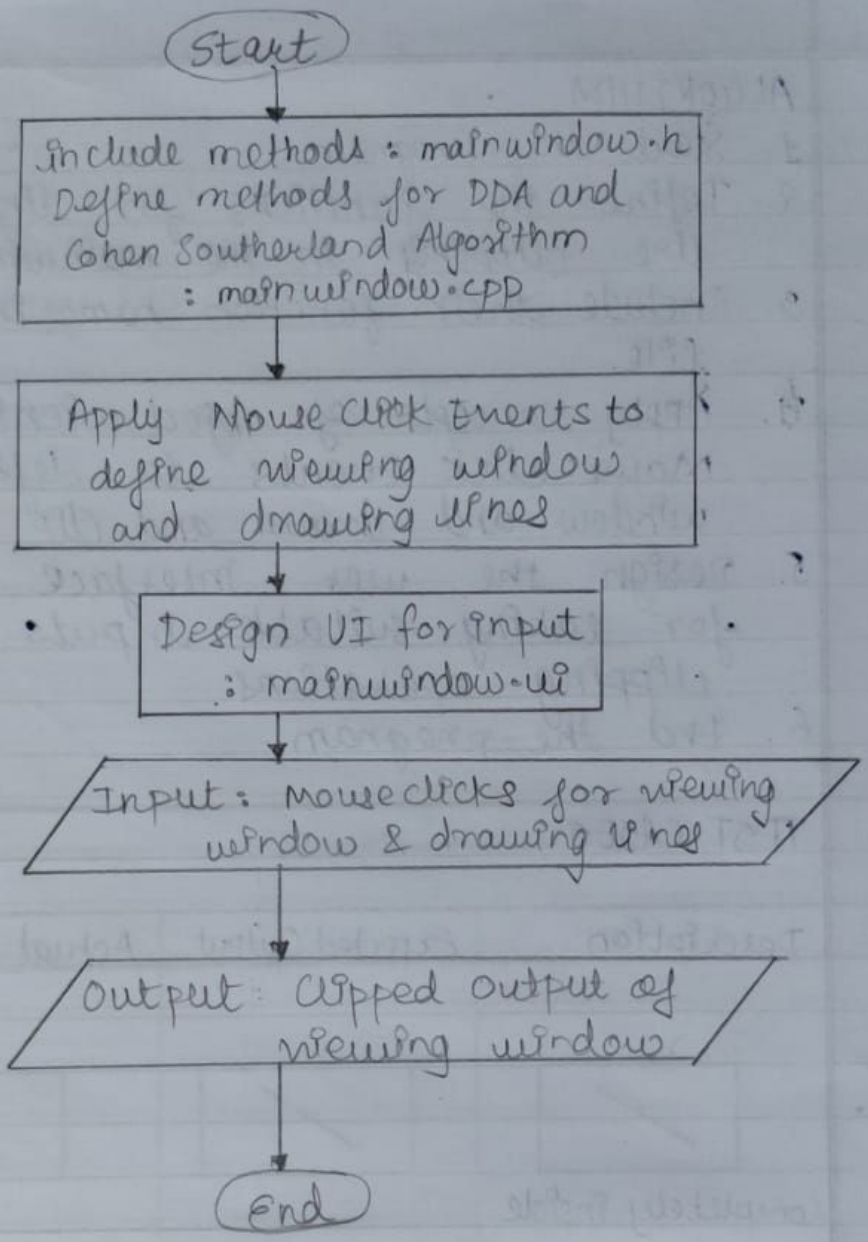
ALGORITHM:

1. Start the Program.
2. Define the functions for line drawing and line clipping in the mainwindow.cpp file.
3. Include their function names in the mainwindow.h file.
4. Apply concepts of object oriented program and Mouse click Events to define a viewing window and draw and clip lines in it.
5. Design the user interface in mainwindow.ui for taking suitable inputs and performing clipping operations.
6. End the program.

TEST CASES:

Sr. No.	Description	Expected Output	Actual Output	Status
1.	 Completely inside			Pass
2.	 Completely outside			Pass
3.	 Partially inside			Pass

FLOWCHART:


Start
Include methods : mainwindow.h
Define methods for DDA and Cohen Southerland Algorithm
: mainwindow.cpp
Apply Mouse click Events to define viewing window and drawing lines
Design UI for input
: mainwindow.ui
Input : Mouse clicks for viewing window & drawing lines
Output : Clipped output of viewing window
End

Rajdhani
DATE / /

CONCLUSION:

The concepts of Cohen Southerland Line Clipping algorithm was successfully understood and implemented using concepts of object oriented programming in computer graphics.

ze Tools Window Help

mainwindow.h <Select Symbol> Windows (CRLF) Line: 1, Col: 1

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>

QT_BEGIN_NAMESPACE
namespace Ui { class MainWindow; }
QT_END_NAMESPACE

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    MainWindow(QWidget *parent = nullptr);
    ~MainWindow();
    void mousePressEvent(QMouseEvent *);
    bool start;
    bool line;
    void DDA(float, float, float, float, QRgb);
    void DDA2(float, float, float, float, QRgb);
    void clipLine(float, float, float, float, QRgb);
    int top = 8,bottom = 4,right = 2,left = 1;
    int outcode(int, int);
private slots:
    void on_pushButton_clicked();
    void on_pushButton_2_clicked();

private:
    Ui::MainWindow *ui;
};
#endif // MAINWINDOW_H
```

1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results

Tools Window Help

main.cpp <No Symbols> Windows (CRLF) Line: 1, Col: 1

```cpp
#include "mainwindow.h"

#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    w.show();
    return a.exec();
}

```

1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results

3 - Qt Creator

Tools Window Help

mainwindow.cpp <Select Symbol> Windows (CRLF) Line: 1, Col: 1

```cpp
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include<iostream>
#include<QMouseEvent>
#include<QtDebug>
#include<QColorDialog>
MainWindow::MainWindow(QWidget *parent)
    : QMainWindow(parent)
    , ui(new Ui::MainWindow)
{
    ui->setupUi(this);
}
MainWindow::~MainWindow()
{
    delete ui;
    start = true;
    line = false;
}
QImage img(500,500,QImage::Format_RGB888);
QImage img2(500,500,QImage::Format_RGB888);
int a[2], b[2];
int r[2], s[2];
float xl, xh, yl, yh;
static int i = 0;
static int f = 0;

void MainWindow::mousePressEvent(QMouseEvent *ev)
{
    int p = ev->pos().x();
    int q = ev->pos().y();
    if(start){
        if(i==0){
            a[0] = p; b[0] = q;
            i++;
        }
        else{
            a[1] = p; b[1] = q;
            if(a[0]>a[1])
            {
                xh = a[0]; xl = a[1];
```

1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results

t3 - Qt Creator

Tools Window Help

mainwindow.cpp | <Select Symbol> | Windows (CRLF) | Line: 1, Col: 1

```
            a[0] = p; b[0] = q;
            i++;
        }
        else{
            a[1] = p; b[1] = q;
            if(a[0]>a[1])
            {
                xh = a[0]; xl = a[1];
            }
            else
            {
                xh = a[1]; xl = a[0];
            }
            if(b[0]>b[1])
            {
                yh = b[0]; yl = b[1];
            }
            else
            {
                yh = b[1]; yl = b[0];
            }
            DDA(a[0],b[0],a[1],b[0],qRgb(225,0,0));
            DDA(a[1],b[0],a[1],b[1],qRgb(225,0,0));
            DDA(a[1],b[1],a[0],b[1],qRgb(225,0,0));
            DDA(a[0],b[1],a[0],b[0],qRgb(225,0,0));
            DDA2(a[0],b[0],a[1],b[0],qRgb(225,0,0));
            DDA2(a[1],b[0],a[1],b[1],qRgb(225,0,0));
            DDA2(a[1],b[1],a[0],b[1],qRgb(225,0,0));
            DDA2(a[0],b[1],a[0],b[0],qRgb(225,0,0));
            ui->label->setPixmap(QPixmap::fromImage(img));
            ui->label_2->setPixmap(QPixmap::fromImage(img2));
            i++;
            start = false;
        }
    }
    if(line){
        if(f==0){
            r[0] = p; s[0] = q;
            f++;
        }
        else{
```

1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results

mainwindow.cpp @ 21425_CG_Assignment3 - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

- 21425_CG_Assignment3
  - 21425_CG_Assignment3
  - Headers
    - mainwindow.h
  - Sources
    - main.cpp
    - mainwindow.cpp
  - Forms

Welcome | Edit | Design | Debug | Projects | Help

mainwindow.cpp | <Select Symbol> | Windows (CRLF) | Line: 1, Col: 1

```cpp
            start = false;
        }
    }
    if(line){
        if(f==0){
            r[0] = p; s[0] = q;
            f++;
        }
        else{
            r[1] = p; s[1] = q;
            DDA(r[0],s[0],r[1],s[1],qRgb(225,225,225));
            ui->label->setPixmap(QPixmap::fromImage(img));
        }
    }
}

void MainWindow::on_pushButton_clicked()
{
    line = true;
}
void MainWindow::on_pushButton_2_clicked()
{

    clipLine(r[0],s[0],r[1],s[1],qRgb(0,225,0));
    ui->label_2->setPixmap(QPixmap::fromImage(img2));
    f = 0; line = false;
}
void MainWindow::DDA(float x1, float y1, float x2, float y2, QRgb col){
    float dx, dy, length, Xinc, Yinc;
    dx = (x2 - x1); dy = (y2 - y1);
    if (abs(dx)>abs(dy)){
        length = abs(dx);
    }
    else{
        length = abs(dy);
    }
    Xinc = dx/length; Yinc = dy/length;
    for(int i = 0;i <length; i++){
       img.setPixel(x1,y1,col);
       x1 = x1 + Xinc; y1 = y1 + Yinc;
    }
```

214...nt3 | Debug

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 6 General Messages | 8 Test Results

ent3 - Qt Creator

ze Tools Window Help

mainwindow.cpp <Select Symbol> Windows (CRLF) Line: 1, Col: 1

```
}
int MainWindow::outcode(int x,int y)
{
    unsigned int code = 0;
    if(y>yh)
    {
        code=code|top;
    }
    if(y<yl)
    {
        code=code|bottom;
    }
    if(x>xh)
    {
        code=code|right;
    }
    if(x<xl)
    {
        code=code|left;
    }
    return code;
}

void MainWindow::clipLine(float x1, float y1, float x2, float y2, QRgb col)
{
    int line_1,line_2,line_n;
    int accept=0;
    int done=0;
    line_1 = outcode(x1,y1);
    line_2 = outcode(x2,y2);
    do{
        if(!(line_1 | line_2))
        {
            accept=1;
            done=1;
        }
        else{
            if(line_1 & line_2){
                done = 1;
            }
            else{
```

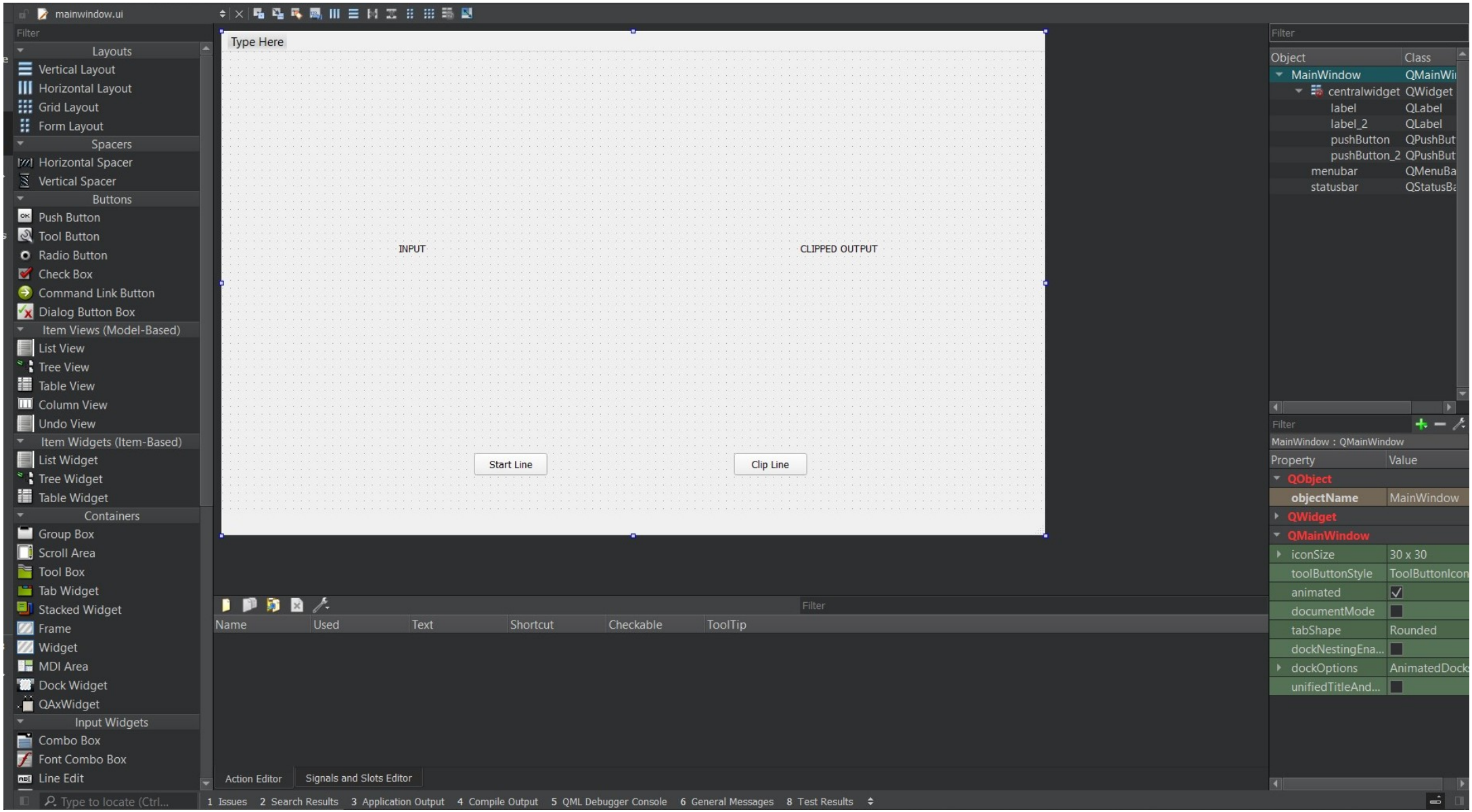
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results

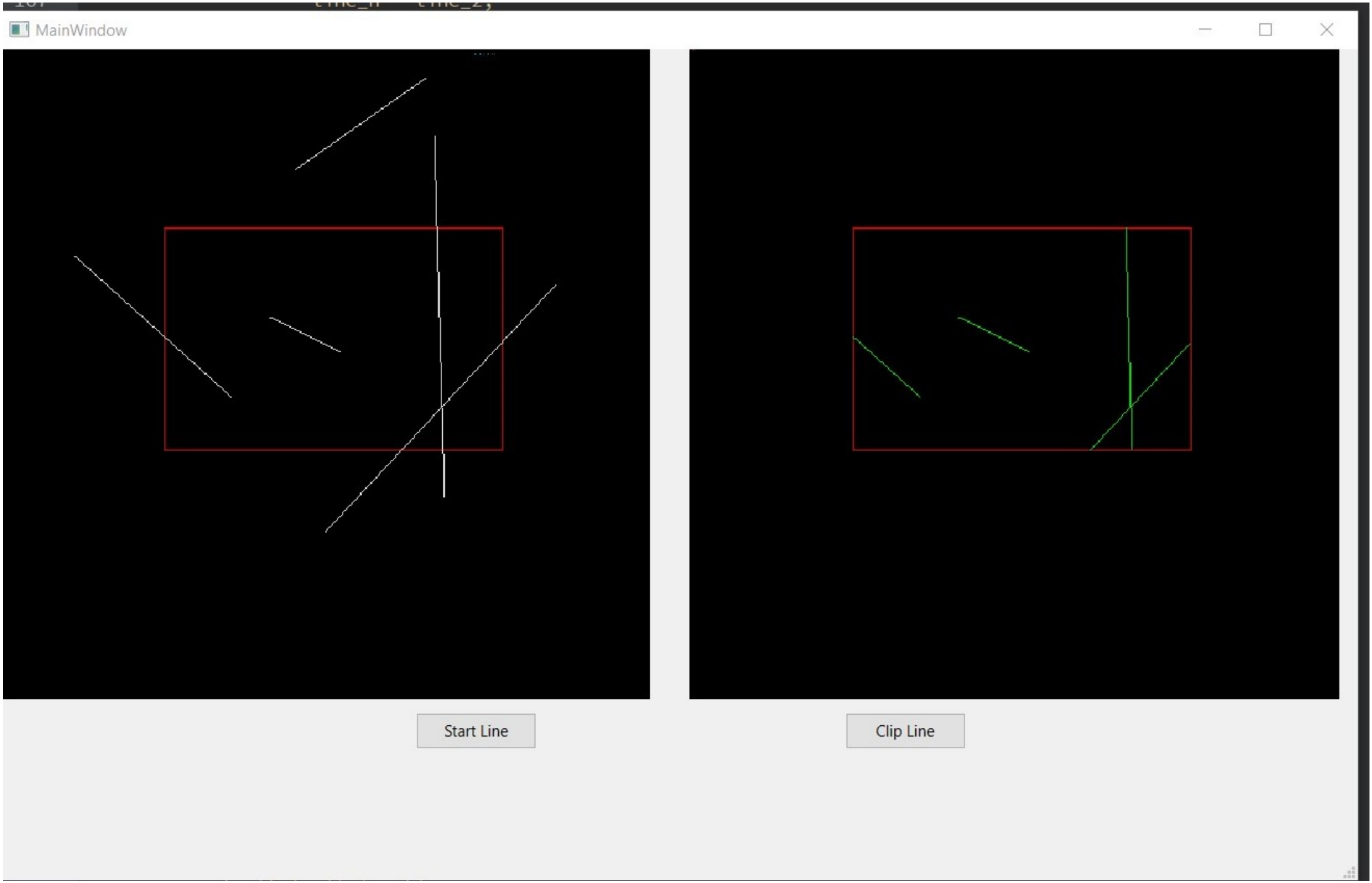
Tools Window Help

mainwindow.cpp | <Select Symbol> | Windows (CRLF) | Line: 1, Col: 1

```cpp
void MainWindow::clipLine(float x1, float y1, float x2, float y2, QRgb col)
{
    int line_1,line_2,line_n;
    int accept=0;
    int done=0;
    line_1 = outcode(x1,y1);
    line_2 = outcode(x2,y2);
    do{
        if(!(line_1 | line_2))
        {
            accept=1;
            done=1;
        }
        else{
            if(line_1 & line_2){
                done = 1;
            }
            else{
                float x,y;
                if(line_1){
                    line_n = line_1;
                }
                else{
                    line_n = line_2;
                }
                if(line_n & top){
                    y=yh;
                    x=x1+((x2-x1)*(yh-y1))/(y2-y1);
                }
                else if(line_n & bottom)
                {
                    y=yl;
                    x=x1+((x2-x1)*(yl-y1))/(y2-y1);
                }
                else if(line_n & right){
                    x=xh;
                    y=y1+((xh-x1)*(y2-y1))/(x2-x1);
                }
                else{
                    x=xl;
                    y=y1+((xl-x1)*(y2-y1))/(x2-x1);
```

1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 6 General Messages 8 Test Results


mainwindow.ui
Filter
Layouts
Vertical Layout
Horizontal Layout
Grid Layout
Form Layout
Spacers
Horizontal Spacer
Vertical Spacer
Buttons
Push Button
Tool Button
Radio Button
Check Box
Command Link Button
Dialog Button Box
Item Views (Model-Based)
List View
Tree View
Table View
Column View
Undo View
Item Widgets (Item-Based)
List Widget
Tree Widget
Table Widget
Containers
Group Box
Scroll Area
Tool Box
Tab Widget
Stacked Widget
Frame
Widget
MDI Area
Dock Widget
QAxWidget
Input Widgets
Combo Box
Font Combo Box
Line Edit
Type to locate (Ctrl...
Type Here
INPUT
CLIPPED OUTPUT
Start Line
Clip Line
Filter
Name
Used
Text
Shortcut
Checkable
ToolTip
Action Editor
Signals and Slots Editor
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
6 General Messages
8 Test Results
Filter
Object
Class
MainWindow
QMainWi
centralwidget
QWidget
label
QLabel
label_2
QLabel
pushButton
QPushBut
pushButton_2
QPushBut
menubar
QMenuBa
statusbar
QStatusBa
Filter
MainWindow : QMainWindow
Property
Value
QObject
objectName
MainWindow
QWidget
QMainWindow
iconSize
30 x 30
toolButtonStyle
ToolButtonIcon
animated
documentMode
tabShape
Rounded
dockNestingEna...
dockOptions
AnimatedDock
unifiedTitleAnd...


MainWindow
Start Line
Clip Line