Technical Manual: Graph Plotter

Date: 25 January 2014

Prepared By: Vrinda Kochar

Contents

1 Introduction

1.1 Document Identification

This document describes the design of Graph Plotter. This document is prepared for assessment in Software Engineering Lab (CS243).

1.2 Application Overview

Graph Plotter plots points in the 2D plane i.e. the x and y coordinates of points on graph and construct lines between those points. The user can change the display properties of the graph and gives the points to be plotted. It is developed as a Microsoft Visual Studio 2010 project, using Visual Basic as the programming language.

1.3 Document Overview

This report details the design process of the Graph Plotter. It first describes the whole application as a whole before going into detail for each individual function. Test carried out are also detailed.

1.4 Reference Documents

The following references have been used:

- http://msdn.microsoft.com
- http://www.vbtutor.net/index.php/visual-basic-2010-tutorial/ : Visual Basic Tutorial

2 Application Description

This section is intended to give a general overview of the basis for the Graph Plotter application design, of its classes, functions, variables and of its development and implementation.

2.1 Introduction

The project "Graph Plotter" is developed as a Microsoft Visual Studio 2010 project, using Visual Basic as the programming language. The project Graph Plotter will plot points in the 2D plane i.e. the x and y coordinates of points on graph and construct lines between those points. The user can change the display properties of the graph.

The report aims to explore it in detail.

2.2 Application Requirements

The project is a Windows Application and requires .NET Framework.

2.3 Architecture

The project uses 2 forms, namely form1 and form2.

Form1.vb imports System.IO as default inputs for Graph Properties are taken from an Input file named "GraphInput.txt". It declares a Data Table variable, "dtGraph", this is used to store the x and y coordinates of points entered by the user, Booleans bgC and gridC that remember whether user changed colours for graph background and grid respectively, ColorDialogs cDialog1 and cDialog2 for graph background color and grid color selection.

The various other toolbox options used in form1 and their uses are:

- DataGridView named "DataGridView1": Has 2 columns "X-Coordinate" and "Y-Coordinate". The user enters the numerical values of the x and y coordinates of the points to be plotted in this DataGrid.
- Labels
- TextBoxes namely:
  - "TextBox1": Minimum of the X-Axis Range
  - "TextBox2": Maximum of the X-Axis Range
  - "TextBox3": Minimum of the Y-Axis Range

    - “TextBox4”: Maximum of the Y-Axis Range
    - “TextBox5”: Interval between two markings on X-Axis
    - “TextBox6”: Interval between two markings on Y-Axis

These textboxes take their default values from the input file “GraphInput.txt”, when the form loads.

- Button named “Button1”: The Button is Green in color and reads “Plot Graph”. Whenever user makes any changes in the DataGridView1 the Button changes color to Red and reads “Redraw Graph”.
- Button named “Button2” reads “Click to choose Graph’s Background Color”. On clicking it opens up a Color Dialog Box which enables the user to select a background color for the Graph. Also the Button’s color is the same as the Graph Background Color.
- Button named “Button3” reading “Click to choose Graph’s Grid Color” enables the user to select a grid color for the Graph. Also the Button’s color is the same as the Graph’s Grid Color.

The functions included in the Form1 class are:

- ReadLine(ByVal lineNumber As Integer, ByVal lines As List(Of String)) As String
  Returns line numbered “lineNumber” from the List of lines “lines”.

The subjects included in this Form1 class are:

- Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
  Form1’s, labels, textbox’s display properties and default values are set and textbox default values read from input file “GraphInput.txt” with the help of “Readline” function, if the file is not found it is created and then values are loaded into the form.
- DataGridView1_CellValidating(ByVal sender As Object, ByVal e As System.Windows.Forms.DataGridViewCellValidatingEventArgs) Handles DataGridView1.CellValidating
  Checks whether value added is numeric, within range of axes(if not changes axes range accordingly) and duplicate x-coordinates are not allowed by popping up a message to the user.

- Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
  Calls Form2, stores the data in the DataGridView1 in dtGraph, sets graphs displays properties as per the inputs taken by the form1. Plots the points taken from dtGraph on the Chart in Form2 and constructs lines between the points. The Graph's background and grid colours are set as default if the user doesn't select a colour (Button2 and Button3 respectively can be used to change colour) else if colour is selected by the user then the same is used in the Graph.
- DataGridView1_CellValueChanged(ByVal sender As System.Object, ByVal e As System.Windows.Forms.DataGridViewCellEventArgs) Handles DataGridView1.CellValueChanged
  Changes the Button1 color and text.
- Button1_LostFocus(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.LostFocus
  Resets the Button1 color and text.
- TextBox1_Validating(ByVal sender As Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles TextBox1.Validating
  Checks whether input is numeric and if so can accommodate the points to be plotted given by the user, in case any of the conditions is not satisfied the user is shown a message giving the details of the same.
- Similarly all other textboxes are also validated.
- Button2_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button2.Click
  Opens a ColorDialog named "cDialog1". If the user selects a color, the Boolean bgC is set to true and the chosen colour is used as the Button colour.
- Button3_Click(ByVal sender As Object, ByVal e As System.EventArgs) Handles Button3.Click
  Opens a ColorDialog named "cDialog2". If the user selects a color, the Boolean gridC is set to true and the chosen colour is used as the Button colour.

Form2 is for displaying the graph plotted with the data from the previous form.

It contains a Chart option named "Chart1" that displays the results when the form is called after user clicks "Button1".

2.4 Input

- User Input:
  The user has to specify the points to be plotted in the DataGridView1, and can change the display properties of the Graph which are shown alongside the Data Grid.
- File Input:
  The default values for the Graph's Display Properties are taken from the text file "GraphInput.txt" which if doesn't exist is created by the program itself and values loaded from it thereafter.
- Input Constraints:
  As the project aims to introduce graphs to 3rd and 4th grade students these constraints have been put to make it easier for them to understand the basic notion of graphs, the way to plot points and join them in increasing order of their x-coordinates to get the graph.
  The points to be plotted should have integer coordinates; display properties should also be numerical. Also the points should be given in increasing order of their x-coordinates.
  In case input file is modified the format of the file should remain the same and only integer values should be replaced.

2.5 Output

The output is the Graph plotted using the data points given by the user, and display properties from form1. The graph is shown in form2, which is called whenever the user clicks Button1.

3 User-Interface Design

The user is presented with a window that requires him to fill in the display properties he desires for the graph, these include:

- Range of X-Axis
- Range of Y-Axis
- Interval on X-Axis
- Interval on Y-Axis
- Background Color of Graph
- Grid Color of Graph

The various points to be plotted are to be written into the Data Grid under the appropriate column (i.e. x-coordinate under column named “X-Coordinate” and y-coordinate under column named “Y-Coordinate”) and then the user clicks on the “Redraw Graph” button.

If any of the inputs given by the user is invalid he is shown a message about the same.

On clicking the “Redraw Graph” or “Plot Graph” button another window shows the resultant graph on the basis of the inputs from the previous window.

4 Application Performance

The Application has been checked for many test cases including points in the 2nd, 3rd and 4th Quadrants of the 2D plane, even though its users (3rd and 4th grade students) will not require these when introduced to graphs for the first time, as the 1st Quadrant is used mostly to teach them.

5 Known Errors

After numerous efforts, I have not been able to remove an erroneous output.

Whenever only a single point is to be plotted with its x-coordinate = 0, i.e. a point on the Y-Axis the output graph shows the point at x=1 and correct input y-coordinate even though the label reads x=0 and the correct input y-coordinate.

So, the Y-Coordinate is correct but the X-Coordinate is changed to 1.

This error has been seen only when a singleton point is plotted and not otherwise.