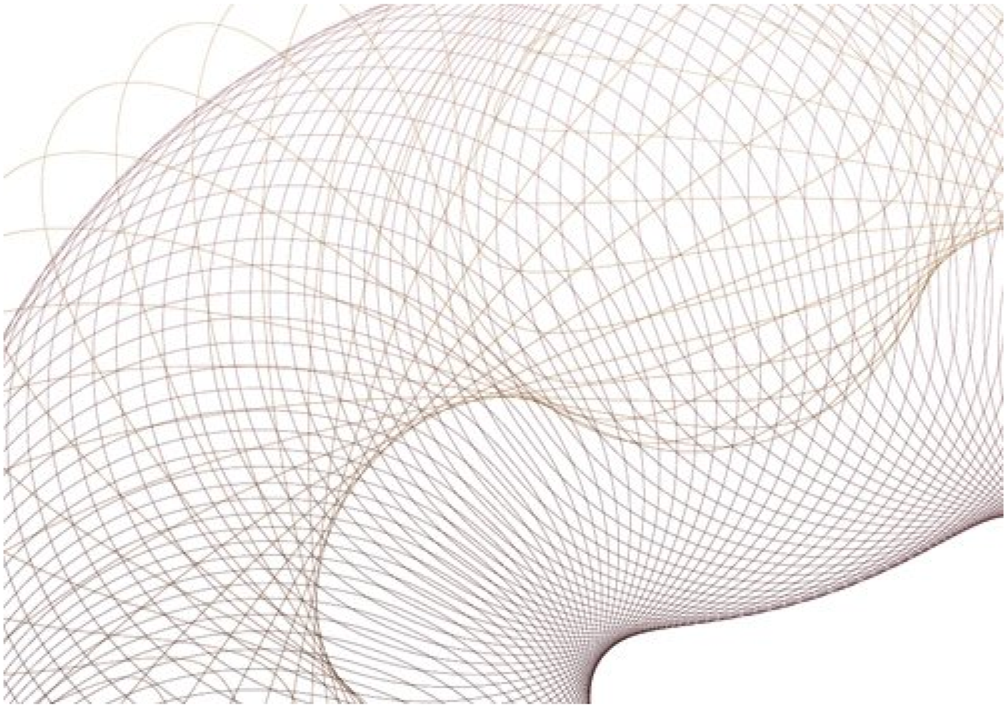
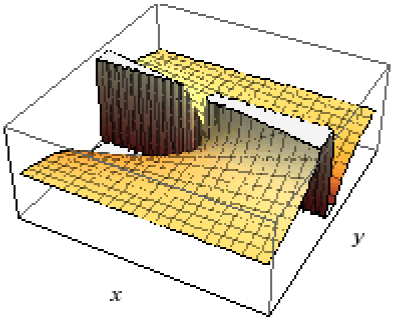
“Generic Function Plotter” 1.0





SUBMITTED BY:

MUHAMMAD AHMED RIAZ

2012-10-0035

CS 200

INSTRUCTOR:

Naveed Arshad

TEACHING ASSISTANT:

Saad Ahmed

ABSTRACT

This project is a standalone application for plotting 2-D function plots.

INTRODUCTION

In this project, I have implemented a graph plotter that can plot functions in Rectangular/ Polar coordinates. It can plot Parametric Functions in both these coordinates systems. It can ensures a good user friendly interface and a range of features that range from zooming to auto scalling the plotted functions.

PURPOSE

The purpose of this project is to provide an easy to handle, multi-purpose function plotter that would give the user a clear concept of how the the function looks. The user can zoom into a particular are to get a clear understanding of how the function behaves in different regions.

FEATURES

The main feature that distinguishes this project from a simply function plotter is that it takes any generic function as input, that might have as many terms and each term may be totally generic

The key features provided by the program are following:

MODES OF PLOTTING

There are two main ploting modes that are further subdivided into two modes each.

1) Polar Plots
2) Rectangular System Plots

In each of these modes, the user may select the simply mode or the Parametric mode. The parametric mode provides the full control over Horizontal as well as Vertical scale.

INPUT MODES

The user, in providing input may enter a function such as sin or cos or arctan. These function may take the input in radians or degrees. When the user gives a range of 0-10, he gets the option to specify that if the given range is in radians or degrees. This input mode becomes really important when the function is composed of functions as sins/cosines as well as normal function (e.g x^2*sinx).

ZOOMING

As the user plots many function on the same canvas(without clearing it) the function may get out of the range of the canvas or the user might want to observe the behavior of the function more closely. This can be achieved by using the option:

ZOOM IN/OUT

ZOOM IN/OUT (x only)

ZOOM IN/OUT (y only)

Thus the user is provided the full control over which direction and axis, the user wants to ZOOM.

AUTO-SCALING

There is a built-in zooming function but the user might not want to press ZOOM-IN/OUT again and again to get the whole picture on the screen. There is an AUTO-Scale function available, that observes all the plotted function, checks what is the maximum range along x/y axis and then automatically scales the function so that all the information is shown on the screen and no information goes out of the screen. This also takes care of the infinities and mostly removes the error and goes not take the scale up to the infinity but intelligently selects the scale so that the infinity part is eliminated. This is really helpful when plotting functions such ad tan(x) which go to infinity in normal range.

RESET-SCALE

Although the Auto-Scale takes care of the infinities and does not take the scale up to that high values in most of the cases, but in some cases, you may observe such problems. After such a scale has been set, using the ZOOM IN option. U can never reach the normal range from infinity. To eliminate this issue there is a RESET-Scale option that brings the scale back to normal range

BUILT-IN FUNCTIONS

You may use many of the mathematical function in defining your input. The complete list of the available function is:

Sin(x)
Cos(x)
Tan(x)
Sinh(x)
Cos(h)
Tanh(x)
Log(x)
Ln(x)
Inverses of the above functions
Abs(x)
x!
Sqrt(x)

NOTE: For the syntax of how to enter thse function, see the INSTRUCTION/HELP page inside the function.

ENGINEERING ALGORITHM

The functions are plotted in the following steps:

PARSING INPUT

The given input is taken in as a string and sent to a class for breaking it up into computer's understandable form.

The steps involved in this process are:

1) Breaking the input into various terms and treating each term separately.
2) Read a term from the left side, if a constant is found, multiply it with the VALUE TO RETURN.
3) If some 'x' is found, look for its power, and add appropriate values to the VALUE TO RETURN.
4) If some function like sin, cos is found, multiply it with the VALUE TO RETURN.
5) If a bracket if found, apply all the steps stated above ,to this bracket separately, and then multiply the result with this term's VALUE TO RETURN.
6) Once all the function are performed, the given class have a method that can take a value of 'x' and return the corresponding value of 'y'.

GIVING RANGE OF x AND FUNDING SCALE

Once we have developed a function that takes ‘x’ and returns ‘y’, this function is given a range of ‘x’ values, (according to the range provided by user) and get the corresponding values of the ‘y’. These values of x and y are studied and their maximum values are found. Using these values, an appropriate scaling factor is determined in order to view all the function within the screen size.

PLOTTING

Once the scalling factor is known, again the range of x is given to the function and corresponding values of y are found and plotted.

ZOOMING

If the user presses the ZOOM button, the scale factor is multiplies with an appropriate number and the old graphs are deleted and they are redrawn using the new scale determined.

PLOTING MODES

If the user has selected the Polar mode, the output is not just ‘y’ ploted against ‘x’, but it is such that the values returned by the function are said to be r and the values given the function are theta. Then the x and y are determined by the formula

x = r*cosine (theta)

y = r*sine (theta)

INPUT MODES

If the user selects the Degree mode, the values provided to the function to calculate y are multiplies by PI/180. Similarly the values returned by Inverse Trigonometric Function are multiplied by 180/PI.

DASH ALL

This function changes all the previously drawn function into dotted lines to differentiate them from the function that are going to plotted after that event.

FUTURE EXPANSION

The main future expansion of this program is to plot 3-D graphs and give the user the option to just click in a part of graph that would automatically ZOOM IN to that part of the graph. This feature wasn’t implemented due to limitation of STROUSTRUP library as is does not offer any MOUSE CLICKING EVENT.