

# PROJECT REPORT

---

*LAB BATCH: 332*

*PROJECT: GRAPH PLOTTER*

*TEAM: Jash Banker (team leader)( 110260003)*

*Keshav Kumar Jha (115060013)*

*Rohan Kora (110010058)*

*Manchella.ch.Sai Hemanth(110010038)*

*Ladi Dinesh Kumar(11D170023)*

*Ramcharan Teja(110040108)*

*Jai Narain(115090018)*

*TA: Ankit Ramteke*

Graph Plotter:

This project aims to plot the graph of various functions. It includes several extra features like plotting the graph of the integral, graph of the derivative and finding the area under the curve and the definite integral of the chosen function in a particular range.

Parts of the project:

- 1) Input: We have selected a list of functions supported by cmath and asked for the choice of function by the user. We have

implemented this through a graphical interface. The user selects a function and his choice is stored in a global variable which is accessed throughout the rest of the program. The input graphic interface is one of the highlights of our program as it involved a lot of effort and debugging to implement the mouse-click feature.

- 2) Checking of range: Since we have given the user the liberty to choose his desired range it is necessary for us to validate the given range. We have written a function to check for the valid ranges of the various functions provided.
- 3) Plotting of the graph: The actual plotting of the graph (after getting the choice of function and getting a valid range from the user) consists of plotting the values of 'y' against the values of 'x'. The value of 'x' is slowly incremented by a small amount, and the adjoining points are joined by a line. As this process is completed throughout the range, we get a smooth graph for our function.
- 4) Scaling of the graph and choice of axes: A very important feature in our program is the scaling of the graph based on the range provided by the user and also based on the maximum and minimum values of the function in the given range. This is done to ensure that the user gets the most relevant choice of scale for a good sized graph. We have also chosen our quadrants based on the sign of the 'x' and 'y' values in the given range so the user gets a zoomed in view of the quadrant(s) in which the graph lies. The scale is automatically decided and the axes are labelled appropriately.

## Extra features

5) Graph of the derivative of the function: We have provided the option to the user to plot the graph of the derivative of the chosen function. This was done using the first principle of differentiation, which gives us the value of the derivative of the function at a particular point. The formula used was  $(f(x+h)-f(x))/h$  with  $h$  being a very small increment depending on the range provided by the user.

6) Graph of the integral of the function: This process is similar to the graph of the derivative. We use Reimann Integration to judge the value of the integral of the function at a particular value of 'x'. Using the obtained value we plot the graph of the function for a range.

We can have infinite parallel curves which are all valid graphs for the integral of the function. We have chosen to plot the graph which passes through the origin.

7) Definite integral and area under the curve: Definite Integral also consists of simply applying Reimann Integral for the given range. This is done by dividing the graph into small strips, finding the algebraic area of each small strip and summing them up. In case of area under the curve, we take the modulus of the acquired algebraic area for each small strip and add them up.

8) Level curves: Level curves are projections of 3D curves onto 2D plane. This can be imagined as slicing the curves using planes that are parallel to x-y plane and viewing this from the top. This is done by equating the  $z=f(x,y)=k$ ; where  $k$  is a constant that can be varied.

In the level curves part of the project we take the choice of function from the user and store it in a variable  $fx$  using switch. Similarly we take  $fy$ . Then we use switch to operate on  $fx$  and  $fy$  and calculate the value using loops. This value is returned as  $f(x,y)$ . Then a loop is run and all values of  $(x,y)$  satisfying  $f(x,y)=k$  in the given range of  $x$  &  $y$  are passed on for plotting.

### Status of completion:

- 1) We have a few errors in our code to validate input. We have missed out on some check conditions.
- 2) Plotting of the graph when the function tends to infinity is not possible. But we have not checked the input for points where the function tends to infinity. As a result no error message is displayed and the graph prints randomly when  $y$  goes to infinity.
- 3) Plotting of the graph of the integral of the function: Although the code for this function works perfectly, the plotting of the graph is really slow as the computer takes a lot of time to compute the value of the integral.
- 4) Level curves: Although we have written a code to find out the relevant points, we were not able to implement the necessary graphics due to lack of time.
- 5) We wished to implement  $\log_{10}(x)$  and  $\exp(x)$  as well but we found a bug in this part at the last minute and did not have sufficient time to decode it.

### Future prospects:

- 1) We would like to work around a way to plot the graph of the function even when it tends to infinity.
- 2) We would like to make the plotting of the graph of the integral of the function faster so that it's feasible for the user.

- 3) We would like to implement level curves as well
- 4) We can try to include composite functions also.
- 5) We can also try to get input from the user in the form of a string so that he/she will have more options in the selection of the function.
- 6) We would like to correct some of the input validation conditions so appropriate error messages are displayed.
- 7) We would like to handle all interactions with the user through a graphical interface. (currently some inputs are through the terminal).
- 8) We would like to expand the plotter to include many more functions and also would like to create a 'scatter plotter' which reads points (for eg experimental data) from a file and join them by a curve.

# Individual contributions

---

1) Ladi Dinesh Kumar

Roll no: 11D170023

CONTRIBUTION: Wrote code for validating input (i.e checking the range of x). This function called checkinput() checks whether the range of x given for the given function is valid. Gave basic design idea for main window on paper.

2) K. Ram Charan Teja

Roll No: 110040108

CONTRIBUTION: also helped design main window on paper with Dinesh. Also gave basic design of the other windows used in input on paper. Helped Dinesh with the function to validate input.

3) Manchella.Ch.Sai Hemanth

Roll No: 110010038

CONTRIBUTION:

The idea of taking up graph plotter as our project was given by me.

- I have written around 300 lines of code for contour lines and tested and debugged it.
- I had also helped the input team with my ideas of inputting, the outline for input through windows was given by me.
- I had also helped in checking the range and its logic.

- I had also given idea for adding integration and differentiation features in the project.
- I also helped in coordinating the team.
- I also wrote a brief description of the level curves part in the SRS.

4) Keshav Kumar Jha  
Roll No:115060013

#### CONTRIBUTION:

Wrote the entire input code for taking input through the graphic interface. Spent a lot of time and effort in learning and implementing the mouse-click features. Tested and debugged the entire input code. Also integrated the other team-mates programs with the input code and removed further bugs. Also helped Jash with the final stage documentation.

5) Jai Narain  
Roll No: 115090018

#### CONTRIBUTION:

Took part in the first two meetings where it was decided that we would take up graph plotter as our project. Met with team-mates in the last week of the project work and observed and tried to understand their codes.

6) Rohan Kora  
Roll No: 110010058

#### CONTRIBUTION:

I have primarily contributed three functions to the project. Graph of integral, area under the curve and definite integral. Although all three work perfectly, the graph of integral function takes a very long time to get plotted.

I have also tried to help various people in their tasks. I helped my teammate with the logic for level curves, although it finally didn't work out. I tried solving problems for the polynomial input and the checking of range of functions.

I spent substantial time trying to help to put together the main program and in debugging the graphic user interface (although the respective person himself sorted out the problem in a different method).

I also wrote the project report for the team and wrote a description of the integral part in the SRS. I have spent around 14 hours on the project excluded the team meetings that I don't remember the dates and times of.

7) Jash Banker (team leader)

Roll No: 110260003

#### CONTRIBUTION:

Wrote the entire graphics code for plotting the graphs of the function, it's derivative and integral. Also wrote the code for generating the derivative. Spent a lot of time and effort in implementing the automatic scaling and shifting of axes. Wrote almost the entire SRS and submitted it. Also edited the project reports and made handled the entire first and second stage submissions. Also helped Keshav and Dinesh debug their programs. Wrote a back-up code for input just in case Keshav's did not work. Helped in integrating everyone's codes. Coordinated with the TA and kept him informed on our progress.