UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 2

Algorithm Analysis and Flowchart

Submitted by:
Bron, Jhustine A.

Instructor:
Engr. Maria Rizette H. Sayo

July 26, 2025

I. Objectives

Introduction

Data structure is a systematic way of organizing and accessing data, and an algorithm is a step-by-step procedure for performing some task in a finite amount of time. These concepts are central to computing, but to be able to classify some data structures and algorithms as "good," we must have precise ways of analyzing them.

This laboratory activity aims to implement the principles and techniques in:

- Writing a well-structured procedure in programming
- Writing algorithm that best suits to solve computing problems to improve the efficiency of computers
- Convert algorithms into flowcharting symbols

II. Methods

A. Explain algorithm and flowchart

B. Write algorithm to find the result of equation: $f(x) = \begin{cases} -x, & x<0 \\ x, & x \geq 0 \end{cases}$ and draw its flowchart

C. Write a short recursive Python function that finds the minimum and maximum values in a sequence without using any loops

III. Results

A. An algorithm is a step-by-step set of instructions used to solve a problem or perform a task, like finding a number in a list using linear search by checking each element one at a time. A flowchart, on the other hand, is a visual representation of an algorithm using shapes and arrows to show the flow of the process. While algorithms are written in plain text and can be a bit harder to understand, flowcharts make it easier to visualize the logic of a program. Both are important tools in programming because they help in planning and understanding how a program works.

B. **Algorithm:**

1. INPUT x
2. If x < 0
3. Set f = -x
4. Else
5. Set f = x
6. Print f
7. End

Flowchart:

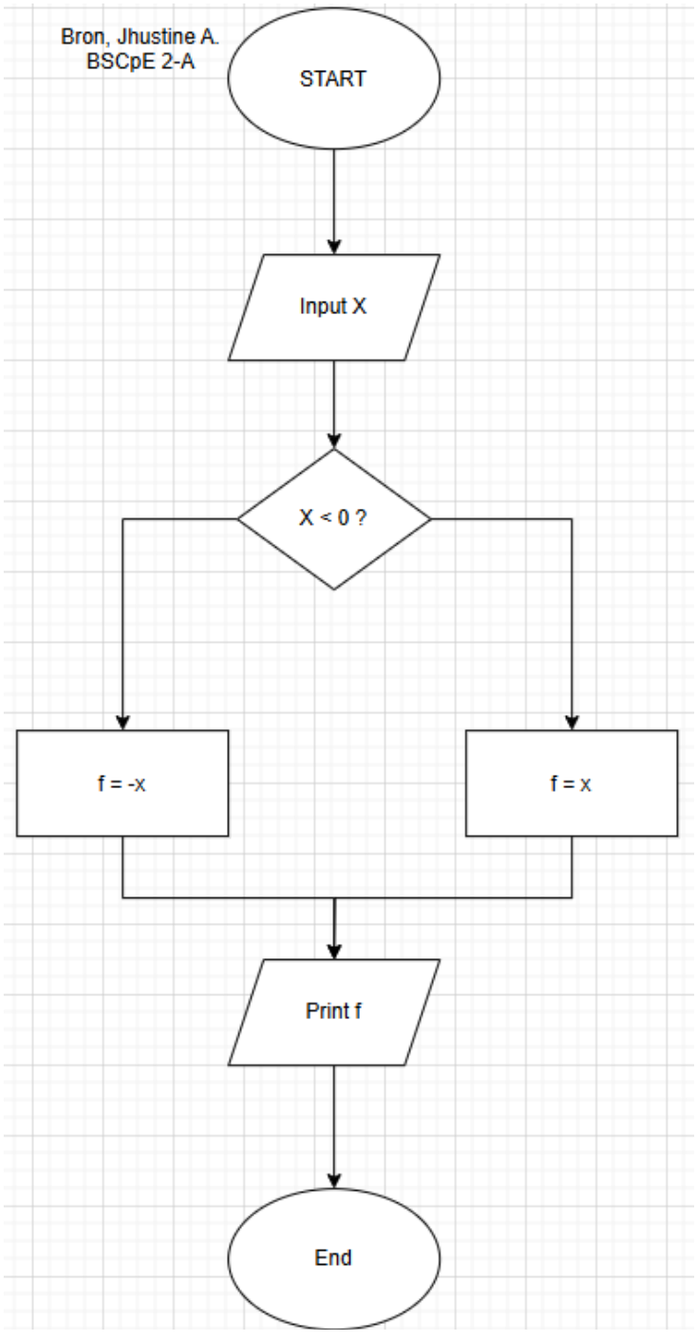

Figure 1. Flowchart of the equation

Insights: this flowchart showcases the basic flow of the given equation, firstly input the x and make a decision whether the value of x is less than 0, if yes, the final value of x will be negative and positive if not.

C. Colab link:
https://colab.research.google.com/drive/1f7WY5oQMaM37Gc4bYdIoQiV0-8_fizFw#scrollTo=0FLztzIeKaP_&line=12&uniqifier=1

Insights:

This code finds the smallest and largest numbers in a list using recursion. If the list has just one number, it returns that number as both the smallest and biggest. If there are more, it checks the rest of the list and compares each one. It then prints out the results in a friendly way.

IV. Conclusion

In conclusion, this activity helped me understand how algorithms and flowcharts work together in solving problems. By writing a simple algorithm and creating its flowchart, I was able to visualize how the steps are followed. I also learned how to use recursion in Python to find the minimum and maximum values in a list without using loops, which shows how powerful and flexible programming can be.

References

GeeksforGeeks. (2025, July 11). *Difference between algorithm and flowchart*. GeeksforGeeks. https://www.geeksforgeeks.org/dsa/difference-between-algorithm-and-flowchart/