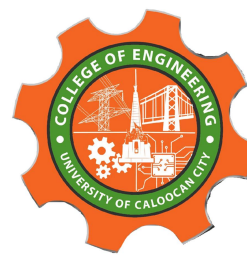




UNIVERSITY OF CALOOCAN CITY  
COMPUTER ENGINEERING DEPARTMENT



Data Structure and Algorithm

Laboratory Activity No. 2

---

# Algorithm Analysis and Flowchart

---

*Submitted by:*  
Directo, Hannah Thea B.

*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 set of instructions that is designed to accomplish a task. Algorithms usually take one or more inputs, run them systematically through a series of steps, and provide one or more outputs. Algorithms are typically associated with computing and are an essential element of computer programming. Flowcharts are visual diagrams that represent data, algorithms, or processes, offering a clear and intuitive way to understand how a program functions. They illustrate step-by-step solutions to problems, making them especially useful for beginner programmers. Additionally, flowcharts play an important role in debugging and troubleshooting by helping identify where issues may occur within a process. Typically, they consist of a series of sequentially arranged boxes that depict the logical flow of operations..

**B. Algorithm**

Step 1: Start

Step 2: Input x

Step 3: If  $x < 0$  then

Set  $f = -x$

Else

Set  $f = x$

Step 4: Output f

Step 5: End

**Flowchart**

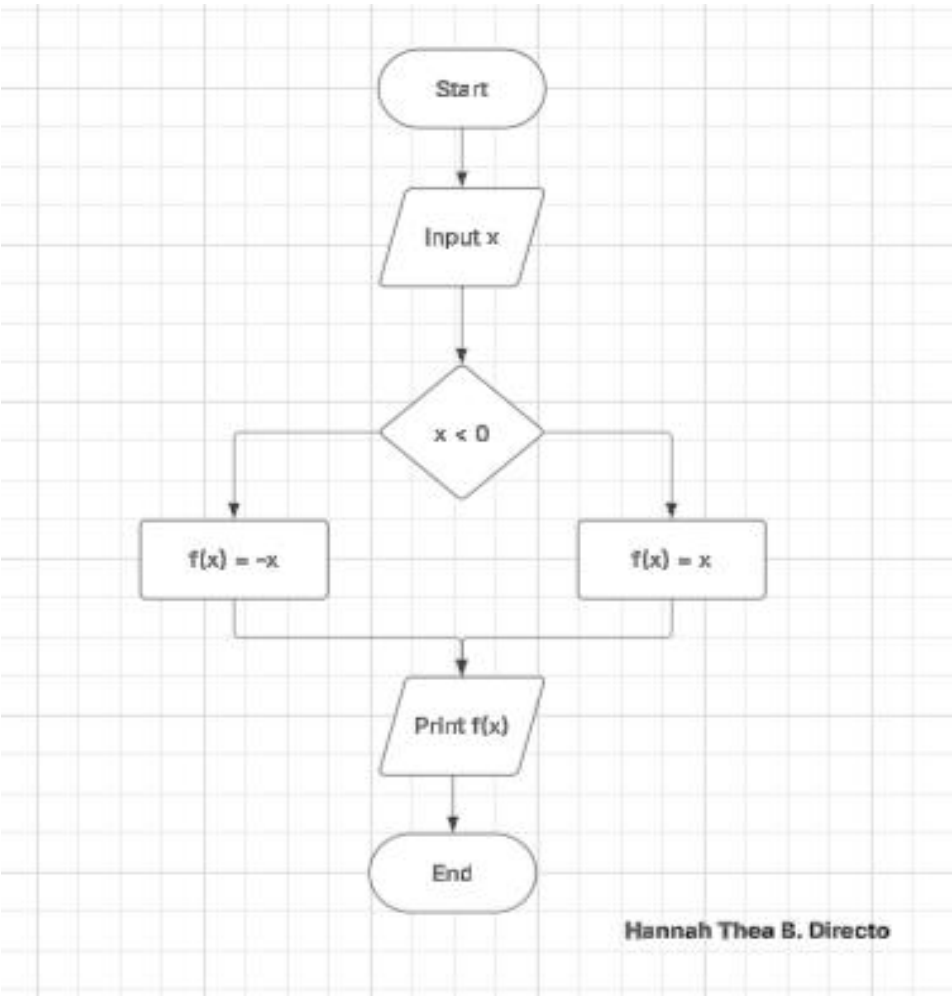
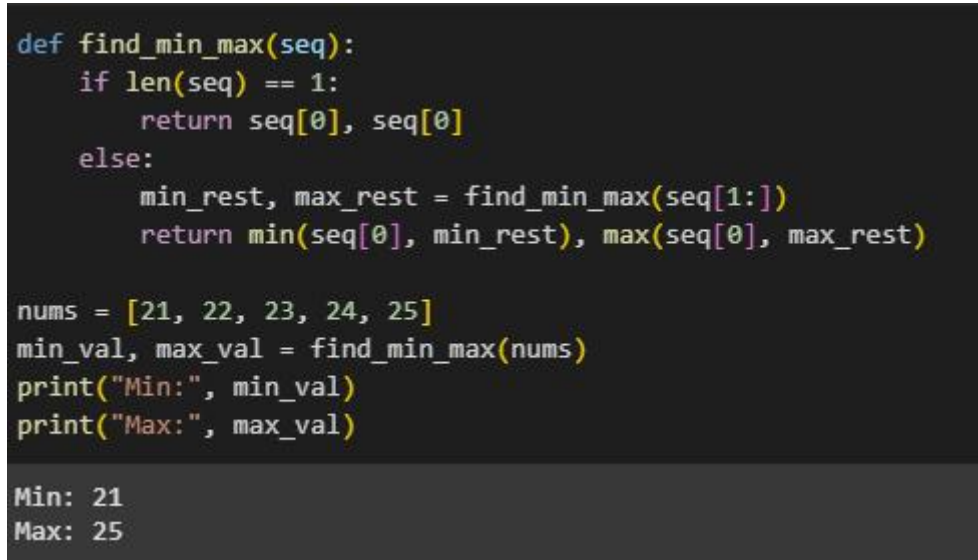


Figure 1 Screenshot of flowchart

The flowchart shows the basic logic of the equation. It starts by taking an input x, then checks if x is less than 0. If it is, the program changes it to -x to make it positive. If not, it keeps the value as it is. After that, it prints the result and ends.

C. Refer to this link:

<https://colab.research.google.com/drive/1OC7QC18qaAq816ZtslpyKva8Td9jTcq0#scrollTo=eTf5SGlV5-Ex>



```
def find_min_max(seq):
    if len(seq) == 1:
        return seq[0], seq[0]
    else:
        min_rest, max_rest = find_min_max(seq[1:])
        return min(seq[0], min_rest), max(seq[0], max_rest)

nums = [21, 22, 23, 24, 25]
min_val, max_val = find_min_max(nums)
print("Min:", min_val)
print("Max:", max_val)

Min: 21
Max: 25
```

Figure 1 Screenshot of colab

This function finds the smallest and largest numbers in a list without using loops by using recursion. It starts by looking at the first number in the list. Then, it calls itself on the rest of the list to find the minimum and maximum values there. After getting those values, it compares them with the first number and updates the minimum and maximum if needed. This process repeats, moving through the list one number at a time, until it reaches the last number

## IV. Conclusion

In this laboratory, we learned how algorithms and flowcharts help us solve problems step by step. We first understood that an algorithm is a list of instructions, and a flowchart is a drawing that shows these instructions clearly. Then, we created an algorithm and flowchart to solve a simple problem that turns negative numbers into positive numbers, which taught us how programs make decisions. Finally, we wrote a Python function that uses recursion to find the smallest and largest numbers in a list without using loops. This activity showed us how organizing steps well helps computers work better and makes programming easier to understand.

## References

- [1] GeeksforGeeks. (2025, April 7). *What is a flowchart and its types?* GeeksforGeeks  
<https://www.geeksforgeeks.org/computer-science-fundamentals/what-is-a-flowchart-and-its-types/>
- [2] Network of the National Library of Medicine. (2022, May 25). *Algorithm*. In *NNLM Data Glossary*. Retrieved August 5, 2025.  
<https://www.nlm.gov/guides/data-glossary/algorithm>