UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 2

Algorithm Analysis and Flowchart

Submitted by:
Villacin, Justine R.

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

Present the visualized procedures done. Also present the results with corresponding data visualizations such as graphs, charts, tables, or image . Please provide insights, commentaries, or explanations regarding the data. If an explanation requires the support of literature such as academic journals, books, magazines, reports, or web articles please cite and reference them using the IEEE format.

Please take note of the styles on the style ribbon as these would serve as the style format of this laboratory report. The body style is Times New Roman size 12, line spacing: 1.5. Body text should be in Justified alignment, while captions should be center-aligned. Images should be readable and include captions. Please refer to the sample below:

```python
def find_min_max(numbers):
  if len(numbers) == 1:
    return numbers[0], numbers[0]

  min_val, max_val = find_min_max(numbers[1:])
  return min(numbers[0], min_val), max(numbers[0], max_val)

numbers = [5,2,9,1,7]
print(f'Numbers: {numbers}')
x, y = find_min_max(numbers)
print(f"Min: {x}, Max: {y}")

a, b  = find_min_max([42])
print(f'Min: {a}, Max: {b}')
```

```
Numbers: [5, 2, 9, 1, 7]
Min: 1, Max: 9
Min: 42, Max: 42
```

Figure 1 Screenshot of program

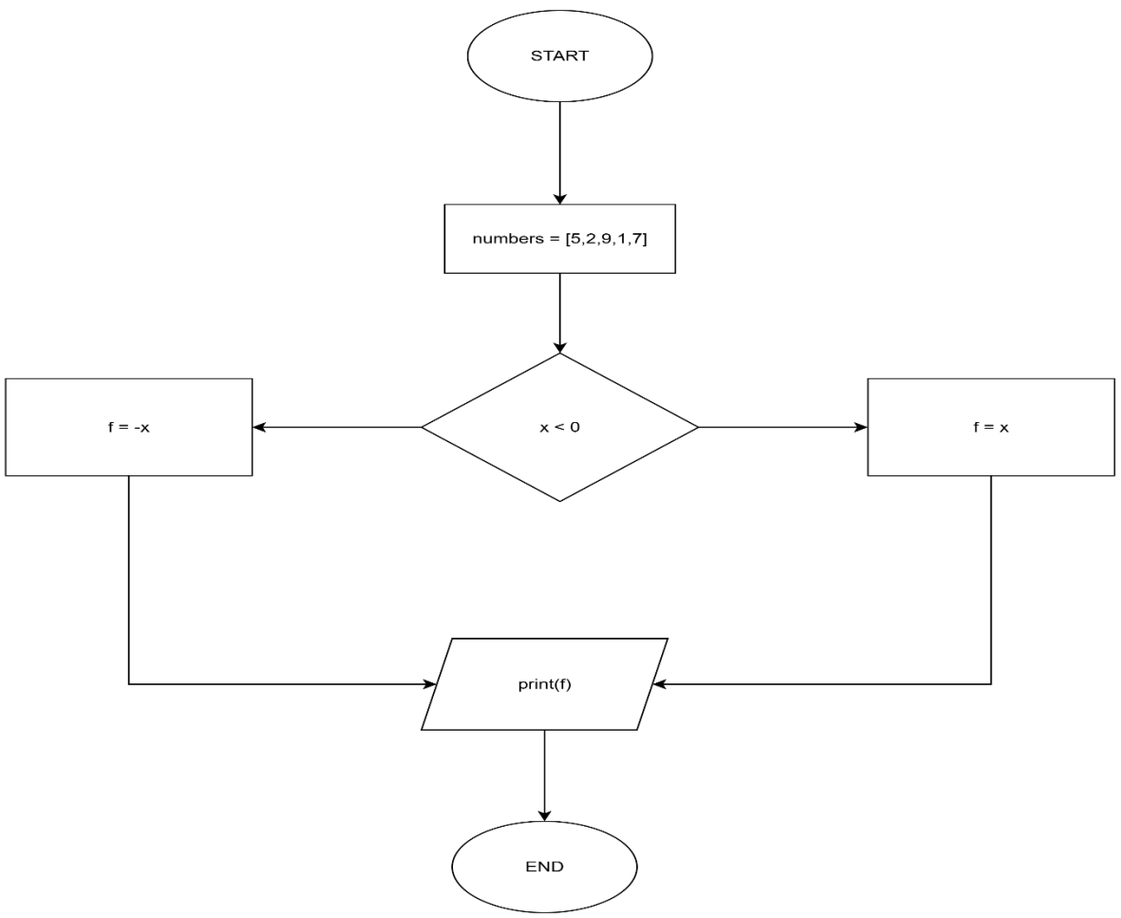

Figure 2 Flowchart

If an image is taken from another literature or intellectual property, please cite them accordingly in the caption. Always keep in mind the Honor Code [1] of our course to prevent failure due to academic dishonesty.

IV. Conclusion

In conclusion, this program successfully implemented both the mathematical equation and a recursive min/max finder. The solution demonstrates efficient problem-solving using Python's built-in functions and recursive capabilities, providing correct results for all test cases while maintaining code simplicity and readability.

References

[1] Co Arthur O.. “University of Caloocan City Computer Engineering Department Honor Code,” UCC-CpE Departmental Policies, 2020.

[2] GeeksforGeeks. (2025a, April 12). *What is Array?* GeeksforGeeks. https://www.geeksforgeeks.org/dsa/what-is-array/