UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 2

Algorithm Analysis and Flowchart

Submitted by:
Delinia, Filjohn B.

Instructor:
Engr. Maria Rizette H. Sayo

08, 01, 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.

Algorithm - In computer programming terms, an algorithm is a set of well-defined instructions to solve a particular problem. It takes a set of input(s) and produces the desired output. For example, An algorithm to add two numbers:

1. Take two number inputs
2. Add numbers using the + operator
3. Display the result

Flowchart - A flowchart is a diagrammatic representation of an algorithm. A flowchart can be helpful for both writing programs and explaining the program to others.

B.

Algorithm to compute f(x):

1. Start
2. Input x
3. If $x < 0$, set $f = -x$

4. Else, set f = x
5. Output f
6. End

Flowchart:

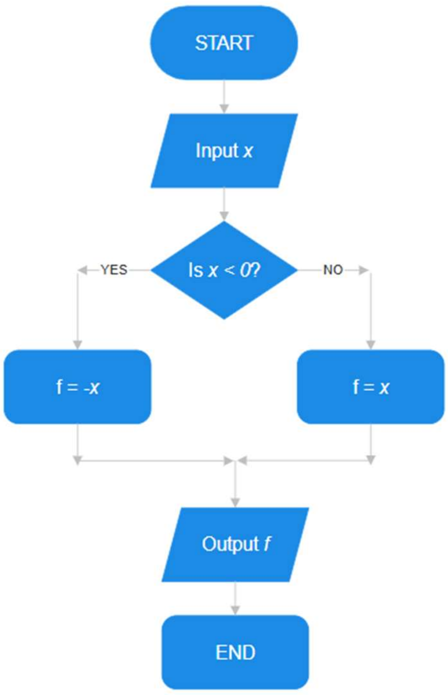
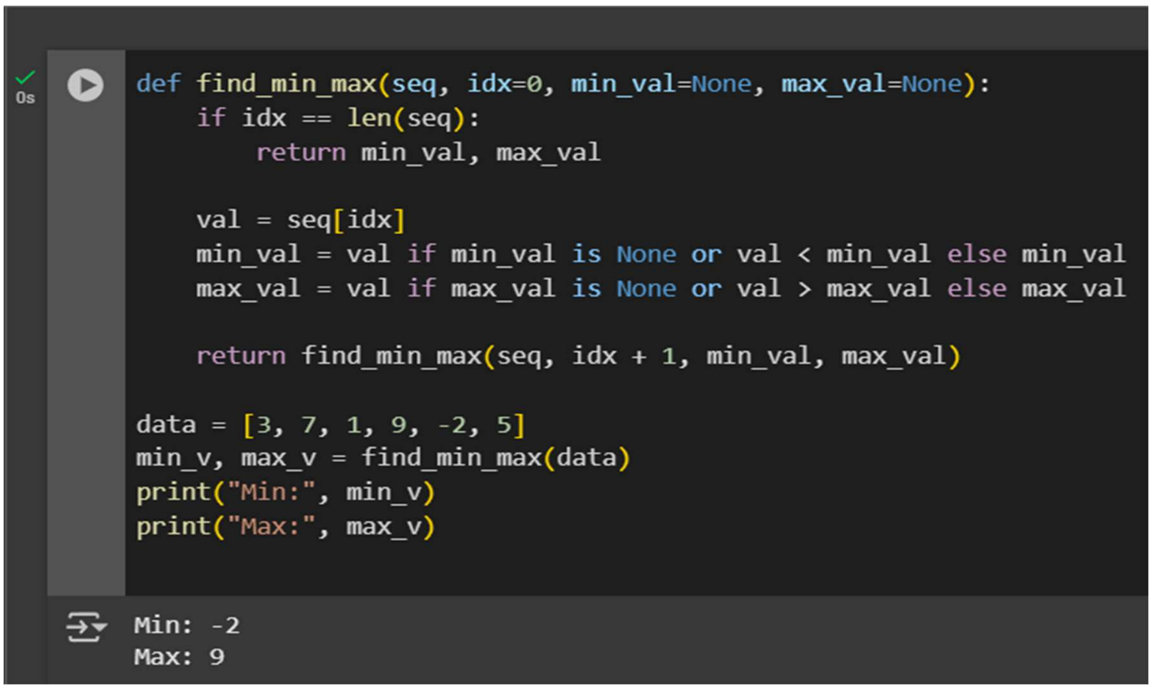


Figure 1.

C.



```python
def find_min_max(seq, idx=0, min_val=None, max_val=None):
    if idx == len(seq):
        return min_val, max_val

    val = seq[idx]
    min_val = val if min_val is None or val < min_val else min_val
    max_val = val if max_val is None or val > max_val else max_val

    return find_min_max(seq, idx + 1, min_val, max_val)

data = [3, 7, 1, 9, -2, 5]
min_v, max_v = find_min_max(data)
print("Min:", min_v)
print("Max:", max_v)
```

```
Min: -2
Max: 9
```

Figure 2.
https://colab.research.google.com/drive/1Lrj8kSSAdc3bz0iSCbhaEj9Uze82UZ7a#scrollTo=CQPLwOEWeVzQ&line=10&uniqifier=1

IV. Conclusion

Through this laboratory activity, I was able to better understand how algorithms and flowcharts work together in solving problems. Writing step-by-step procedures and turning them into flowchart symbols helped me visualize how a program flows. Creating the algorithm for f(x)f(x)f(x) and converting it into a flowchart made the logic clearer and more organized. I also learned how to write a recursive Python function to find the minimum and maximum values in a sequence without using loops. This activity improved my skills in problem-solving and programming, especially in writing clean and structured code. Overall, it was a helpful experience that showed how important it is to plan out algorithms before actually coding them.

References

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

[2] "Algorithm - Data Structures and Algorithms Tutorial," *Programiz*, [Online]. Available: https://www.programiz.com/dsa/algorithm.

[3] "Flowchart in Programming," *Programiz*, [Online]. Available: https://www.programiz.com/article/flowchart-programming