UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 3

Translating Algorithm to Program

Submitted by:
Barbas, Steven Jade P.

Instructor:
Engr. Maria Rizette H. Sayo

August, 02, 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 tasks 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
- Writing an efficient Python program from translated algorithms

II. Methods

- Design an algorithm and the corresponding flowchart (Note: You may use LucidChart or any application) for adding the test scores as given below if the number is even: 26,49,98,87,62,75
- Translate the algorithm to a Python program (using Google Colab)
- Save your source codes to GitHub

III. Results

Algorithm :

- Start
- Initialize a list of test scores
- For each number in the list:

  If the number is even, add it to total
- Print the value of total
- End

Flowchart:

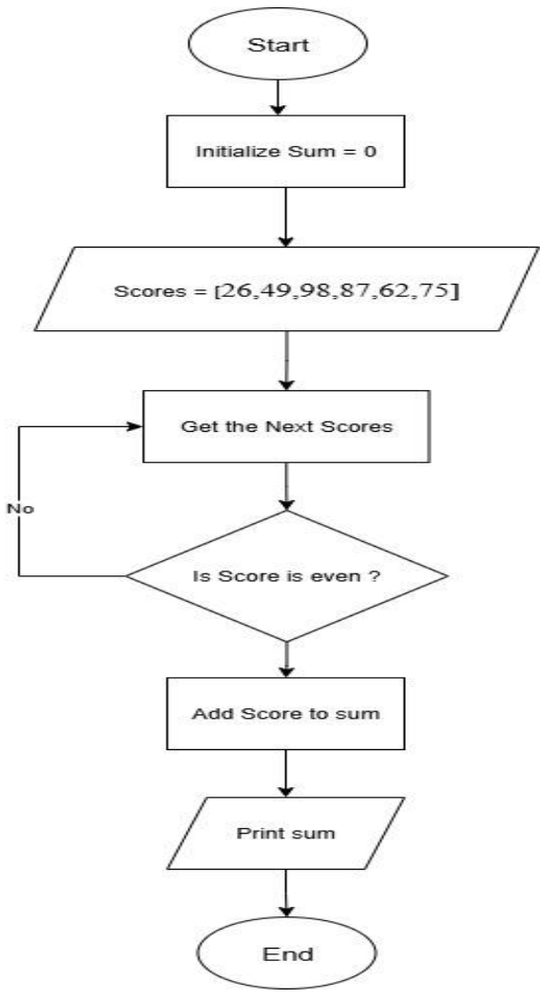
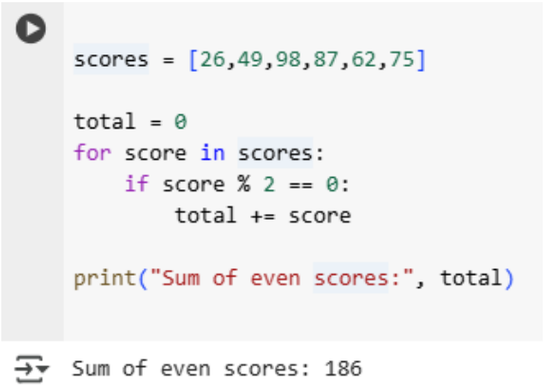


Figure 1 Flowchart

This flowchart illustrate the process of calculating the sum of even number from the list of scores. It begins with a "Start" symbol, followed by initializing a variable called sum to 0. The list of scores is defined as [26, 49, 98, 87, 62, 75]. The program then enters a loop where it gets the next score from the list one by one. For each score, it checks if the number is even by determining if it is divisible by 2. If the score is even, it is added to the sum. If not, the program skips the number and moves to the next one. This process continues until all scores in the list have been checked. After finishing the loop, the program prints the total sum of the even numbers, then ends.

Source Code:



```
scores = [26,49,98,87,62,75]

total = 0
for score in scores:
    if score % 2 == 0:
        total += score

print("Sum of even scores:", total)
```

```
Sum of even scores: 186
```

Figure 2 Screenshot of Sourcecode

Link of Sourcecode: LAB-3 - Colab

This Python program adds up all the even numbers from a list of scores. It starts by setting the total to 0. Then, it goes through each number in the list one by one. For each number, it checks if it is even by using score % 2 == 0. If the number is even, it adds it to the total. After checking all the numbers, it prints the final sum. For example, in the list [26,49,98,87,62,75], the even numbers are 88 and 56, so the output is: "Sum of even scores: 186".

IV. Conclusion

This activity helped me gain a clearer understanding of how algorithms, flowcharts, and code are interconnected. Starting with a basic task adding only the even numbers from a list, I was able to break it into simple steps, represent it visually through a flowchart, and then turn it into working Python code. Seeing how each part contributes to the overall process made it feel more practical and easier to understand. It showed me that even straightforward exercises can be very educational when approached step by step.