SLIIT ACADEMY

FCIT – Semester 1

DEVELOPING AN ALGORITHM

Ovini Seneviratne

Learning Outcomes

End of this lecture you will be able to learn ,

LO1 : To define the problem.

LO2 : To write the solution algorithm.

LO3 : To desk check the solution.

1. Defining the problem

- To help with this initial analysis, the problem should be divided into three separate components:
  - Input : a list of the source data provided to the problem.
  - Output : a list of the outputs required.
  - Processing : a list of actions needed to produce the required outputs.

Defining the problem

- The **input** and **output** components are identified
- The **processing component** is also identified

Example 01 : Add three numbers

Problem - A program is required to read three numbers, add them together and print their total.

Step 1: Underline the **nouns** and **adjectives** used in the specification. This will establish the input and output

Step 2: Underline the **verbs** and **adverbs** used in the specification. This will establish the actions required.

Defining Diagram - Add three numbers

Input	Process	Output

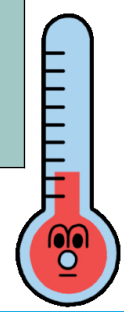
Example 2 : Find average temperature

A program is required to prompt the terminal operator for the maximum and minimum temperature readings on a particular day, accept those readings as integers, and calculate and display the average temperature to the screen.

Average Temperature = (maximum temperature + minimum temperature) / 2.

Defining Diagram – Find Average Temperature

Input	Process	Output



2. Designing a Solution Algorithm

- Most challenging task in the life cycle of a program
- **Pseudocode** is useful in this trial-and-error process, since it is relatively easy to add, delete or alter an instruction.
- A solution algorithm will have sequence control structure, decisions structures or loops.

Solution algorithm - Add three numbers

Solution Algorithm - Find average temperature

Key Features in Solution Algorithm

- **BEGIN** statement - starting point of the algorithm
- **END** statement - indicates that the algorithm is complete.
- All processing steps should be included between the **BEGIN** and the **END** statement.
- **Each processing step in the defining diagram relates directly to one or more statements in the algorithm**

Key Features in Solution Algorithm

Defining Diagram	Algorithm (Pseudocode)
Read three numbers	
Add numbers together	
Print total	

Key Features in Solution Algorithm

Defining Diagram	Algorithm (Pseudocode)
Prompt for temperature	
Get temperature	
Calculate average temperature	
Display average temperature	

3. Checking the Solution Algorithm

- This step is necessary because most major logic errors occur during the development of the algorithm, and **if not detected these errors can be passed** on to the program.
- It is much easier to detect errors in the pseudocode than in the corresponding program code
- **Desk checking** involves tracing through the logic of the algorithm with some chosen **test data**

Desk checking your solution

- **When selecting test data to desk check an algorithm, choose simple test cases.**
- **There are six simple steps to follow when desk checking an algorithm:**
  - Choose simple input test cases that are **valid**. Two test cases are usually sufficient.
  - Establish the expected result for each test case.
  - Make a table on a piece of paper of the relevant variable names within the algorithm.
  - Walk the first test case through the algorithm, line by line, keeping a step by- step record of the contents of each variable in the table.
  - Repeat the walk-through process using the other test data cases.
  - Check that the expected result matches the actual result.

Example 1: Desk check - Add three numbers

```
   BEGIN
1      INT number1, number2, number3 ,total=0
2      PRINT 'Enter three numbers'
3      GET number1, number2, number3
4      total = number1 + number2 + number3
5      PRINT 'Total is ', total
   END
```

	First data set	Second data set

Desk Checking

Statement number	number1	number2	number3	total	Output

Example 2: Desk check - Find average temperature

```
    BEGIN
1       FLOAT max_temp, min_temp, avg_temp
2       PROMPT 'Enter max_temp and min_temp'
3       GET max_temp, min_temp
4       avg_temp = (max_temp + min_temp)/2
5       PRINT 'Average Temperature is ' , avg_temp
    END
```

	First data set	Second data set
max_temp	30	40
min_temp	10	20
avg_temp	20	30

Desk checking

Statement number	max_temp	min_temp	avg_temp	Output

Summary

-
-
-