28

School of Computing and Information Technologies

PROGCON - CHAPTER 2

CLASS NUMBER: 28

SECTION: ABPSY181

DATE:

NAME: Pearl Hope Tabat

PART 2: Identify whether each variable name <u>is valid</u>, and if not explain why.

a) Age

valid

b) age_* $5r\jmath$

invalid. a variable must not contain an arithmetic expression.

c) +age

invalid. a variable must not contain an arithmetic expression.

d) age_

valid

e) _age

valid

f) Age

valid

g) 1age

valid

h) Age 1

invalid. a variable must not have a space or else the program will consi
it as two.

CLASS NUMBER: 27   SECTION: APPSTIU

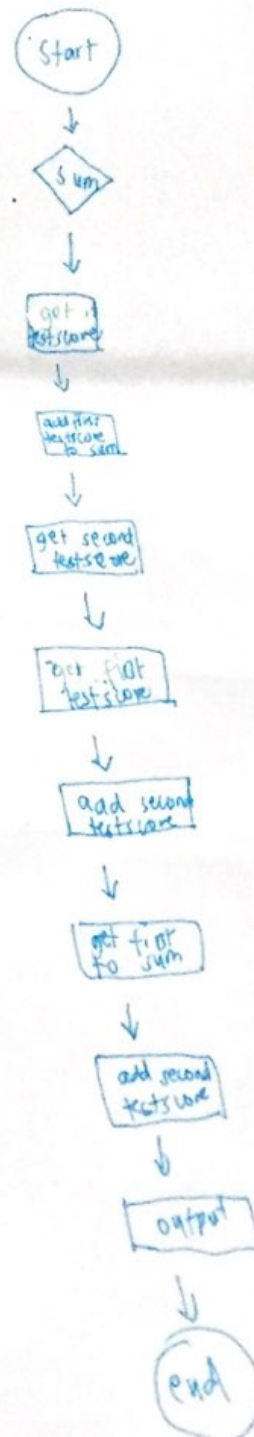NAME: Pearl Hope Jabut   DATE:

Exercise 1. Design **an algorithm** and **the corresponding flowchart** for adding the test scores as given below: 85, 65, 78, 52, 98, 83

a) Algorithm

1.   Start
2.   Sum = 0
3.   Get the first testscore
4.   Add first testscore to sum
5.   Get the second testscore
6.   Add first test score to sum
7.   Get the __first__ testscore
8.   Add to __second__
9.   Get first to sum
10. Add score testscore
11. Get first to sum
12. Add score testscore
13. Get first to sum
14. Add second test Score
15. Output
16. End

b) Draw the corresponding flowchart of the above algorithm

CLASS NUMBER: 27   SECTION: AMPSY1R1

NAME: Pearl Hope Tabora   DATE:

Flowcharting Exercises

Exercise 2: The problem with the algorithm from Exercise 1 is that, some of the steps appear more than once, i.e. step 5 get second number, step 7, get third number, etc.

One could shorten the algorithm as follows:

1. Start
2. Sum = 0
3. Get a value
4. sum = sum + value
5. Go to step 3 to get next Value
6. Output the sum
7. Stop

This algorithm and its corresponding flowchart are a bit shorter than the first one. In this algorithm, step 3 to 5 will be repeated, where a number is obtained and added to sum. Similarly, the flowchart indicates a flow line being drawn back to the previous step indicating that the portion of the flowchart is being repeated.

Draw the corresponding flowchart of the above algorithm.
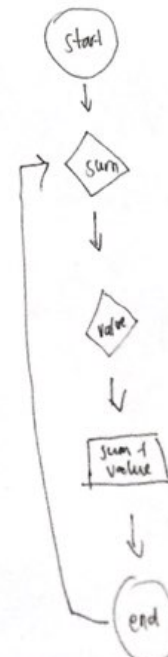
CLASS NUMBER: 27

SECTION: ADIS118I

NAME: Kean Hope tabad

DATE:

Flowcharting Exercises

Exercise 3: From the exercise 2, one problem indicates that these steps will be repeated endlessly, resulting in an **endless** algorithm or flowchart.

The algorithm needs to be improved to eliminate this problem. In order to solve this problem, we need to add a last value to the list of numbers given. This value should be unique so that, each time we get a value, we test the value to see if we have reached the last value.

In this way our algorithm will be a finite algorithm which ends in a finite number of steps as shown below. There are many ways of making the algorithm finite.

The new list of numbers will be 10, 20, 300, 4000, 50000, 1, -1. The value −1 is a unique number since all other numbers are positive.

1. Start
2. Sum = 0
3. Get a value
4. If the value is equal to −1, go to step 7
5. Add to sum (sum = sum + value)
6. Go to step 3 to get next Value
7. Output the sum
8. Stop

Draw the corresponding flowchart of the above algorithm.