(COMP-1400) Lab Exercises #2

Objective:

The main objective of this exercise is to help students to break down a problem into smaller units and write a step-by-step solution for a given problem using a **pseudocode** or **flowchart** diagram.

Part A: a sample pseudocode and flowchart. Try to recreate this flowchart in Raptor (see raptor flowchart at the end of the document). Part A is not graded, however, you can recreate this flowchart and modify it to solve Part B more effectively.

Problem: Get a positive integer value from the input and calculate and print the number of digits.

Sample Input	Sample Output
10	2
150	3
266002	6

Sub-solution (Counting Digit Parts):

```
ALGORITHM CountDigits
INPUT: n      (or READ n)
OUTPUT: count
BEGIN
    count ← 0
    WHILE (n!=0) DO
        n ← n/10
        count ← count+1
    END WHILE
    PRINT count
END
```

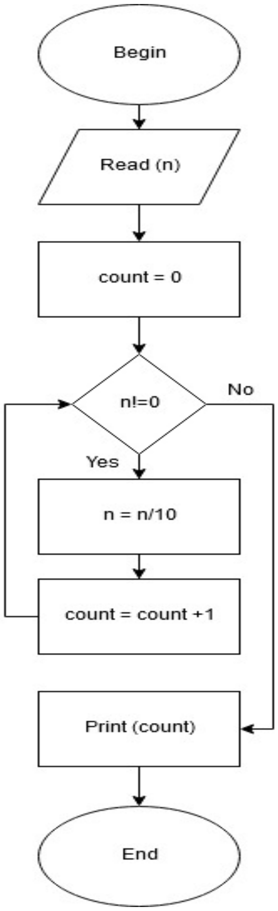
Description:

Assume the given number is n, use a loop statement, and continuously divide the n by 10 until n becomes 0. Count the number of iterations and print it. In fact, the number of iterations represents the number of digits.

Note 1: If you want to know (extract) the last digit of a number, you can always obtain it by (n % 10) which is a reminder of dividing n by 10, e.g., if n is 1234, then (1234%10) is 4.
We name operation % as "mod" or "modulo".

Note 2: By dividing a number, with d digits, by 10, you can get the d-1 most significant digits from the number, e.g., (1234/10) is 123 and similarly (678/10) is 67.
We name this "Integer Division".

(Raptor flowchart on the last page)

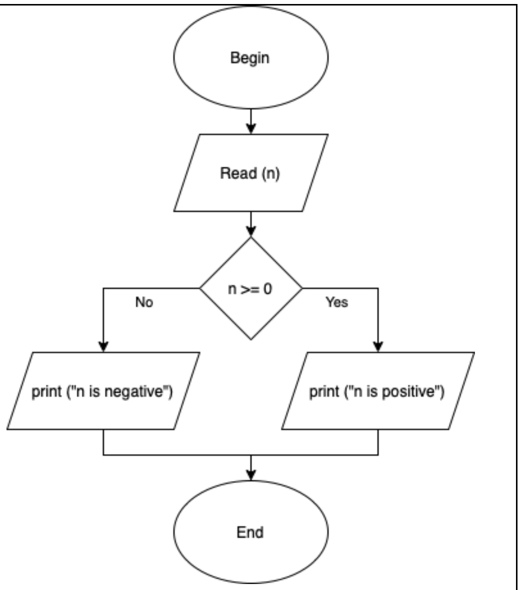


#Sub-solution(to check the positive part): To check if an input number is positive or not, we can consider the following algorithm.

```
ALGORITHM CheckPositive
INPUT: n
OUTPUT: “n is positive”, “n is negetive”
BEGIN
    IF (n >= 0) THEN
        PRINT “n is positive”
    ELSE
        PRINT “n is negative”
    END IF
END
```

Description:

The program reads an integer from the user and determines if the number is positive using ‘n >= 0’ (n bigger than or equal to 0) and prints the result, otherwise this means n must be negative so we print “n is negative”.



#Practice to merge: Now try to combine both sub-algorithms so that the solution for the Problem would print “n is negative” if the user inputs a negative number. The number of digits will be calculated only if the number is positive.

Draw the flow chart based on the following algorithm.

```
ALGORITHM CountDigitsPositive
INPUT: n
OUTPUT: count, “n is negative”
BEGIN
    IF (n <= 0) THEN
        PRINT “n is negative”
    ELSE
        count←0
        WHILE (n!=0) DO
            n ←n/10
            count ← count+1
        END WHILE
    END IF
END
```

Part B: Write a pseudocode (follow a similar format as shown in PART A), and draw a flowchart for the following problems. To design/draw the flowchart, you have the options to do it in different ways, using a flowchart software like Raptor, or a graphical software like "https://www.draw.io", or simply on a piece of paper.
Note: Two Software tools, Raptor and Flowgorithm, are available on CS servers using NoMachine.

Problem 1: Reverse the digits of a number (10 points)

Get an integer number from the user and print the number in the reversed order on the screen. Use math arithmetics to solve this problem (no lists/arrays or strings).

Sample Input	Sample Output
15	51
126	621
266002	200662

Problem 2: Find the frequency of a digit in a number (10 points)

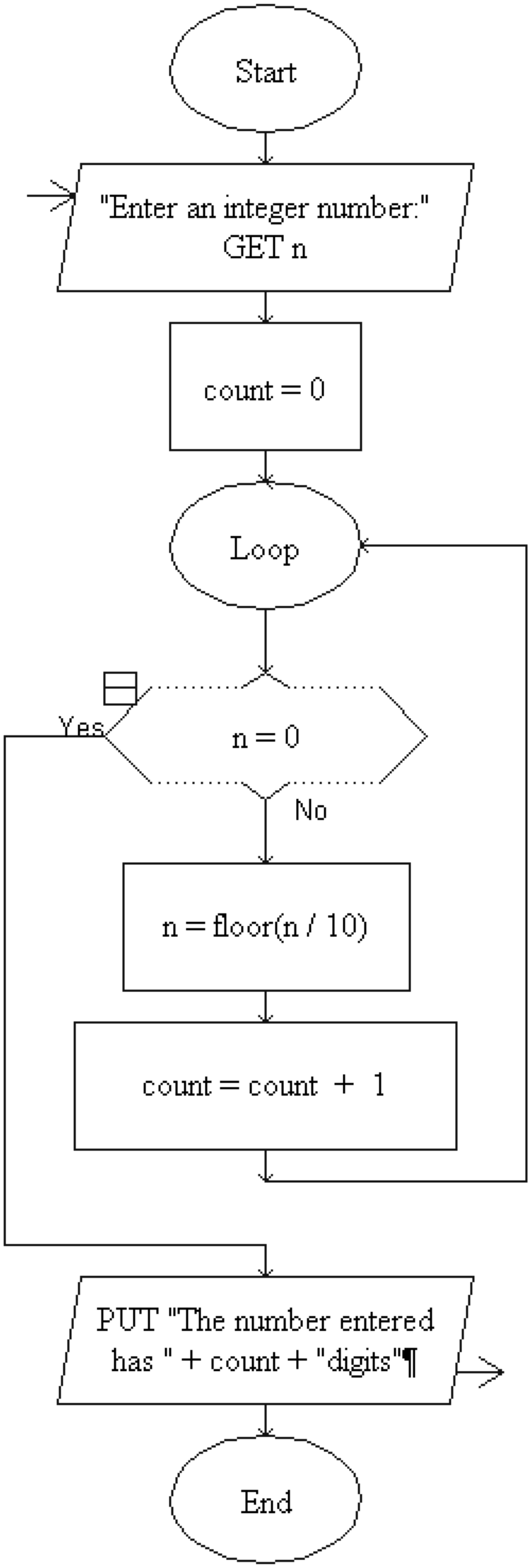
Get an integer number, and a digit from the user and calculate and print out the frequency of the given digit in the number. Use math arithmetics to solve this problem (no lists/arrays or strings).

Sample Inputs	Sample Output
123333 3	4
222545 4	1
555 2	0
922325 2	3

EVALUATION: You need to show your GA/TA the complete pseudocode and/or flowcharts during the lab. Grading is done during lab 2, brightspace submissions are required but not used for grading. GAs/TAs should ask questions about the solutions presented.

Total: 20 marks. (Partial marks can be given for attempts)

A sample Raptor flowchart for Part A


Start
"Enter an integer number:"
GET n
count = 0
Loop
n = 0
Yes
No
n = floor(n / 10)
count = count + 1
PUT "The number entered has " + count + "digits"¶
End