

COSC2406 SS19 – Assembly Language Programming

Assignment 4 - Covers up to Chapter 5 & 6

Due: Thursday August 1st, 2018 by 11:55pm

Good formatting and commenting are essential for assembly program. A penalty of up to 25% will be applied if the programs are not properly formatted and commented.

Complete the following Programming Exercises **USING ONLY MATERIAL UP TO CHAPTER 6:**

FOR EACH QUESTION: It does help to write pseudo-code to identify the steps necessary to solve these problems. Transfer the refined algorithms to your program as comments and then write the Assembly code to perform the necessary steps. It is recommended that this step is done first, and coding second.

If you have any issue with the questions or unclear about the requirements, please email me immediately.

A) [20] Chapter 5 – Exercise 5 (Pg 187) – BetterRandomRange Procedure

Create a test program that asks the user for a lower number and an upper number – and then generate a 10 random number from the lower number to the upper number. The questions should be repeated 5 times before stopping – the loop needs to include asking for the lower and upper numbers. Make sure you use the Randomize library function in your program (it only needs to be called once and usually as the first line of the program).

B) [40] The IsLetter, ToUpperCase and ToLowerCase Procedures

- a) Make a procedure, much the like the IsDigit procedure code shown at the end of Chapter 6, only the IsLetter procedure will return ZF=1 if the value in the AL register is a letter, either upper case or lower case. If the value in AL is not a letter, then ZF will return as ZF=0. The procedure must preserve AL.
- b) The difference between the uppercase letters and the lowercase letters is that bit 5 (20h) is on for the lowercase letters and off for the uppercase letters. Create two more procedures, ToUpperCase and ToLowerCase. ToUpperCase must turn the 5th bit off (use an AND operation) and the ToLowerCase must turn the 5th bit on (use an OR operation).
- c) Make a test program that will do the following:
 - a. Ask the user to enter a string using the ReadString procedure in the Irvine Library. The string will be limited to 250 characters.
 - b. Copy the string to two other arrays.
 - c. Count the number of Letters in the original string and report the count.
 - d. Change all the letters in the first copy to UpperCase.
 - e. Change all the letters in the second copy to LowerCase

COSC2406 SS19 – Assembly Language Programming

- f. Print the count, the uppercase version of the string and the lowercase version of the string.

C) [40] Count Digits and Letters in a text file

Write a program that asks the user for the name of a text file located in the same folder as this program's .asm file (such as another .asm file). Open the file using the OpenInputFile procedure. Include appropriate code to detect if the operation fails and report the error.

Once open, read the file into a buffer, 100 characters at a time. Process the characters such that you could the number of digits and letters. You will want to keep track of how many individual letters and numbers – so you can report how many 0's, 1's, 2's...a's b's etc. that are in the file. I suggest using one or two arrays to hold the count values. You will also need some of the procedures from Question A.

Once the entire file has been processed, print the results in a manner similar to the output shown to the right.

Your program should have two custom procedures: (a) one to process the characters read into the buffer and count each letter and number; and (b) one to print the count results. Both these functions must preserve the registers: one will use USES and the other push/pop operations.

**Please enter the name of the file:
Fibonacci.asm**

**Fibonacci.asm
count of '0' = 3
count of '1' = 8
count of '2' = 7
count of '3' = 1
count of '4' = 0
count of '5' = 0
count of '6' = 0
count of '7' = 0
count of '8' = 0
count of '9' = 0
count of 'A' = 31
count of 'B' = 24
count of 'C' = 24
count of 'D' = 20
count of 'E' = 55
count of 'F' = 20
count of 'G' = 8
count of 'H' = 4
count of 'I' = 30
count of 'J' = 1
count of 'K' = 0
...**

COSC2406 SS19 – Assembly Language Programming

Submission, demonstrating, and grading:

For each of the programs, provide in a Word file (one file only): (i) the headers for each program showing the TITLE, description, your name, assignment & problem number, and the date you wrote the program; and (ii) screen captures of the working programs. The file name must be ***ASSIGN4_YourName.docx*** where YourName is your last name followed by your first name. Upload this file and all of your ***.asm*** files for each of the programs to CMS. (Example: Using my name, the file would be ASSIGN4_LajoieMichael.docx.) Also, be sure to include your name at the beginning of the Word document. This document will contain:

- i. The headers for each of your solutions
- ii. A screen shot (using the snipping tool) showing the program output.

All programs must be fully commented using the two column approach discussed in class and shown throughout the textbook in the code examples. ***Failure to comment the code properly or to format like the code in the textbook will result in a penalty of up to 25%.***

Demonstrate and explain your working programs to the TA, Johnny Console (giconsole@algonau.ca) by Tuesday Aug 6th. The demonstration and explanation of your code will represent 25% of the assignment grade – the other 75% of the grade will come from grading the code and how well the programs meet the specification and requirements of the question.

IMPORTANT:

- a) ***If you do not demonstrate your programs to the TA by the end of the demonstration date, you will receive a grade of zero.***
- b) ***If you are unable to explain most or all of your code and the logic of your programs, then the maximum grade possible will be 50%.***