

COSC2406 F18 – Assembly Language Programming

Assignment 4 - Covers up to Chapter 5

Due: Thursday November 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 5:**

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) [10] BetterRandomRange Procedure

Create a procedure that returns a random number between a low number and a high number. The low number will be supplied in ebx and a high number in eax. The random number must have the possibility of returning both the low number and the high number.

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. Make sure you use the Randomize function (part of the Irvine library) in your program; it only needs to be called once and usually as the first line of the main procedure.

B) [30] Populate an array with random numbers

Create a program creates an array of 10 SBYTES which are initially filled with the value 100. Using your BetterRandomRange procedure from part A, populate the array with random numbers between in the range of -50 to +75. Make sure to include RANDOMIZE in your main method.

Create a procedure to print the array which accepts the address of the array in the register ESI, the TYPE of the array in EBX, and the number of elements in ECX. This procedure will print an array of signed integers, separated by 2 spaces and enclosed in curly braces { }. Use this method to print the array both before and after the random numbers are added to the array.

COSC2406 F18 – Assembly Language Programming

Next, create a procedure that will add up all the values in the array. Even though the values of the array are smaller than DWORDS, add up the numbers using the DWORD register EAX and return the total in the EAX register. Like the print procedure, this sumArray procedure will accept the address of the array in the register ESI, the TYPE of the array in EBX, and the number of elements in ECX.

Finally, make any necessary adjustments to your program so that if the array type is changed from SBYTES to SWORDS, the program still works correctly.

C) [20] Upper and Lower Case

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 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). Both of these functions will accept the offset of the string to be converted in the EDX register and the length of the string in the ECX register.

Create a test program that asks the user to enter a message (can be up to 50 characters long – not counting the null terminator). Print the word "ORIGINAL: " followed by the string that was entered. Then convert the string to uppercase using your procedure and then print the word "UPPERCASE: " followed by the converted string. Finally convert the string to lower case and print the word "LOWERCASE: " followed by the converted string.

D) [20] Reading from a File

Create a program which collects the name of a file from the user (read a string), then used the filename to open a text file. Once open, the program will read the contents of the text file, 20 bytes at a time, and print the contents to the console window. When done, make sure you close the file. Add a delay of ¼ second after printing each block of 20 bytes. Create a text file with any content you like to test your program.

COSC2406 F18 – 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, Glenn Driver (gdriver@algonau.ca) by Thursday Nov 8th. 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%.***