

CSE 341
Fall 2018
Project #1

Due: Monday, October 8, 2018 at 11:59 PM

Part 1

Design a program that will prompt the user to enter five decimal numbers. Each number will NOT exceed five digits, may be signed, and will be an integer. The program can NOT ask the user how many digits will be entered. Each number must be stored in memory as a NULL terminated strings. The first number must be stored at the memory address 0x10000000. The second number must be stored at the address 0x10000008. The third number must be stored at the memory address 0x10000010. The fourth at 0x10000018, and the fifth at 0x10000020. The user may enter comma, such as 19,281, when entering the numbers but it is not required. If a comma is entered, it should not be stored as part of the string. The carriage return should not be stored as part of the string. The string should store the minus (-) character before the digits if the number is negative number. The user may or may not enter the plus (+) character for a positive number, but if they do, the plus (+) character should not be stored as part of the string. A string without a plus (+) or minus (-) character will be assumed to be positive. All subroutines must adhere to the MIPS software convention.

Part 2

Write a MIPS assembly language subroutine called *average* that accepts arguments in \$a0, \$a1, \$a2, and \$a3. The first three arguments are the addresses of three NULL terminated strings. The subroutine should convert each of the three strings to signed integers (in two's complement format), average the integers, store the average as a NULL terminated string at the address passed into the routine in \$a3, and return the average (as a signed integer in two's complement format) in \$v0. The string that is stored must adhere to the same specifications (commas should not be stored, carriage return should not be stored, plus sign should not be stored, and a minus sign if applicable must precede the digits stored). Note that the strings representing the numbers being averaged must NOT be modified by the routine! All subroutines must adhere to the MIPS software convention.

Part 3

Write a MIPS assembly language subroutine called *sort* that sorts the five NULL terminated strings stored at addresses 0x10000000, 0x10000008, 0x10000010, 0x10000018, and 0x10000020. ***When sorting, the values to be sorted should be treated as integers, despite the fact that they are represented in memory as strings!*** All subroutines must adhere to the MIPS software convention.

Submission & Grading

Each part must be in its own separate file, named `project1part1.s`, `project1part2.s`, and `project1part3.s`.

Your program must be written using only native MIPS instructions. All assembly code produced in this project will be graded by running the program in SPIM or XSPIM on *timberlake* in bare mode, so make sure that your code works properly in those simulation environments. To invoke SPIM or XSPIM in bare mode, use the command line argument *-bare*. Your *commented* assembly code must be submitted online using the ***submit_cse341 project1part1.s project1part2.s project1part3.s***.