

Gebze Technical University
Department of Computer Engineering
CSE 241/505
Object Oriented Programming
Fall 2016
Homework # 2
Due date Oct 16th 2016

You will continue developing your CPU in this homework. You will have now memory instructions for your CPU. The memory addresses and constraints should be kept different, so we represent memory addresses with a # sign as a prefix. For example 100 is a constant but #100 is an address.

The instructions in HW1 are still valid. Your new instructions are

Move instructions

MOV reg, address

MOV address, reg

MOV address, constant

For example MOV R1, #45 copies the value of register 1 to memory location 45

Addition and Subtraction instructions

ADD reg, address

SUB reg, address

For example, ADD R1, #45 adds the value of memory location 45 to register 1

Print instructions

PRN address

For example, PRN #56 will print the value of the memory location 56 to the screen, after each print a new line should be inserted.

Modified instructions

HLT

halts the program and prints on the screen the contents of all registers as well as the the memory.

Your program will run using the same command line parameters as HW1. However, we have a new option

The format for the command line parameters is as follows

yourProg filename option

yourProg is the name of your executable file, file name is the text file that contains your simple CPU instructions, option a number and defines how your program runs as follows

- if option is 0, your program will run and finish by executing each instruction

- if option is 1, your program will execute each instruction after displaying the instruction first. It also will print the contents of all the registers.
- If option is 2, your program will execute each instruction just like the option 1. This option will also print the contents of the memory after each instruction execution.

Write a CPU program that sort 10 integers from smallest to largest. The integers are 30, 23, 4, 56, 34, 56, 89, 32, 45, 25. Your program should write these integers to the memory using MOV instructions first. You will include your program with your submission.

Important Notes:

- Your memory can hold only 50 integers. Each integer is a single unsigned byte. Addresses start from 0.
- Your program can have at most 200 instructions or lines.
- Your program should handle error cases such as syntax errors in the input files. You should print an error message on the screen and halt the program if you detect an error in the input.
- With your submission, include the results of a few runs of your program with different programs and run options.
- Do not use any functions from the standard C library (like `printf`), you will use `<<` and `>>` operators to print and write strings.
- Do not use any string functions like `atoi` or `strtok` or similar. If you need these functions, then write your own!
- You will use C++ string class to manipulate your strings. You may use any string member functions.
- Your program should have only functions and no classes.
- Do not forget to indent your code and provide comments.
- You should submit your work to the moodle page. You should strictly follow the submission instructions which will be available shortly.