

The UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Comp 411 Computer Organization

Fall 2009

Problem Set #4

Issued Wednesday 10/14/09; Due Wednesday, 10/21/09

**Homework Information:** Late homework will not be accepted. Feel free to get help from others, but the work you hand in should be your own. You may **not** use solutions to a previous year's homework to aid you in completing this assignment. Please review the honor code for programming assignments on the class website.

This assignment consists of three parts. For the first two parts, you will be writing functions to be called by other assembly programs. Since other programmers may be using your code, you **must** manage the stack appropriately by following all the conventions discussed in class regarding saved and temporary registers, passing arguments and values, etc. Your final step in this assignment is to combine these two functions into one program.

**Problem 1. ASCII string to binary number (ps4atoi.asm)**

Write a function `atoi` that converts a string of up to 8 ASCII digits into a 32-bit integer. The function will receive as an argument the starting address of the string and must return a 32-bit integer containing the integer value of the string. Assume that the string is an ASCIIZ string, i.e., ends with the null character (ASCII code 0). You don't need to check for errors in the string, i.e., you may assume the string contains only characters '0' through '9', and will not represent a negative number. For example, `atoi` called with the argument "12345" will return 12345.

**Problem 2. Computing Fibonacci numbers (ps4fib.asm)**

Write a **recursive** function `fib` that computes the Nth Fibonacci number,  $Fib(N)$ , where  $Fib(0)=0$ ,  $Fib(1)=1$ , and  $Fib(n)=Fib(n-1)+Fib(n-2)$ . This function takes in a single unsigned 32-bit integer and will return an unsigned 32-bit integer result. You can assume that the computed Fibonacci number will fit within 32 bits (unsigned). A non-recursive function will get no credit.

**Problem 3. Put it all together (ps4complete.asm)**

Once you have created complete and correct functions for Problems 1 and 2, incorporate them into the provided assembly file for part 3. This main procedure should call `atoi` on the provided string in order to convert the string value to an integer, then call your `fib` function to compute the corresponding Fibonacci number. Finally, the main procedure will print out the result.

**Submission:** Starter assembly programs are provided for you similar to last assignment. Remember to prefix each one with your onyen (e.g., <your-onyen>-ps4fib.asm) When you have completed your assignment, email all files and a screenshot of the final output, with subject "<your-onyen> PS4 Submission" to:

**unc.comp411@gmail.com**

**Grading:** Problem 3 will be graded first, and if it is correct, you will receive full credit. If incorrect, problems 1 and 2 will be graded separately (40 points each, no partial credit). Therefore, possible grades for this assignment are 0, 40, 80, and 100.