

The UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

Comp 411 Computer Organization

Spring 2011

Lab #7: More Recursion

Issued Fri. 2/25/11; Due Fri. 3/2/11 (beginning of lab)

You are to essentially redo Lab #6 but implement a function to calculate the Nth Fibonacci number instead of factorial.

Exercise 1. Computing Fibonacci numbers (fib.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 implementation of the function will receive zero credit.

A starter assembly file **fib.asm** is provided on the class website. For testing purposes, this file has a `main` procedure that calls `fib` repeatedly to compute and print `fib(0)` to `fib(12)`.

Exercise 2. Put it all together

Use your file **complete.asm** from Lab 6, and make any changes needed to work for this lab. Reuse your implementation of `atoi` from Lab 6 (or, if there were errors, you may borrow someone else's or ask the instructor/TA). Copy and paste your implementation of `fib` from Exercise 1 into **complete.asm**. Your `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 program for Exercise 1 is provided for you on the class website. Fill in your name and onyen in the comments at the top. Add your code where indicated; do not change anything else in the file! When you have completed your assignment, submit the following:

- Exercise 1: copy of the code (**fib.asm**) and a picture of the program's output
- Exercise 2: copy of the code (**complete.asm**) and a picture of the program's output.