EECE.3170: Microprocessor Systems Design I

Summer 2016

Homework 4

Due **1:00 PM, Thursday, 6/9/16**

Notes:

- While typed solutions are preferred, handwritten solutions are acceptable
- All solutions must be legible and contained in one file. Archive files are not acceptable.
- Electronic submissions should be e-mailed to Dr. Geiger at Michael_Geiger@uml.edu. Please include your name as part of your filename (for example, mgeiger_hw4.pdf).
- This assignment is worth 100 points.

1. (40 points) Write the following subroutine in x86 assembly:

```
int f(int v1, int v2, int v3) {
   int x = v1 + v2;
   return (x + v3) * (x - v3);
}
```

Recall that:

- Subroutine arguments are passed on the stack, and can be accessed within the body of the subroutine starting at address EBP+8.
- At the start of each subroutine:
  i. Save EBP on the stack
  ii. Copy the current value of the stack pointer (ESP) to EBP
  iii. Create space within the stack for each local variable by subtracting the appropriate value from ESP. For example, if your function uses four integer local variables, each of which contains four bytes, subtract 16 from ESP.
  iv. Local variables can then be accessed starting at the address EBP-4.
- A subroutine's return value is typically stored in EAX.

See Lectures 14 and 16-18 for more details on subroutines, the x86 architecture, and the conversion from high-level concepts to low-level assembly.

2. (60 points) Write the following subroutine in x86 assembly:

```
int fib(int n)
```

Given a single integer argument, `n`, return the `n`th value of the Fibonacci sequence—a sequence in which each value is the sum of the previous two values. The first 15 values are shown below—note that the first value is returned if n is 0, not 1.

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
fib(n)	0	1	1	2	3	5	8	13	21	34	55	89	144	233	377