Homework #1 – 50 pts
Submit your homework to TEACH by Wed. 10/03/2018, at 11:59pm

Your submission should be comprised of two files: a .pdf file containing written answers to the questions and one .c file containing C code. You do not need to submit your assembly code. Legible, handwritten solutions are acceptable for undergraduate students (472). The handwritten solutions must be electronically scanned and submitted as a PDF file. Graduate students (572) must compose their solutions in MS Word, LaTeX, or some other word processor and submit the results as a PDF file.

1. (8 pts) Assume that the signed numbers in this question utilize the two's complement convention.
    a. What is the 12 bit binary representation of -142 (decimal)?
    b. Perform the following equation in 12 bit binary (show your work): 119 + (-142)
    c. What is the smallest number of bits that can correctly represent the **signed binary number** corresponding to 56841 (decimal)?
    d. What is the **unsigned** hexadecimal representation of 56841?

2. (7 pts) Using the 32 bit IEEE 754 standard:
    a. What is the binary representation of 572.375?
    b. What is the most-negative floating point number that can be represented in 32 bit IEEE 754 format (ignoring negative infinity)?
    c. Show the binary representation of the number in part (2b) above.

3. (4 pts) Assuming the decimal number: 472.2
    a. Why is it not possible to 100% accurately represent the number 472.2 in 32 bit IEEE 754 format?
    b. Can this number be represented with 100% accuracy in IEEE 754 double precision format (64 bits)?

4. (4 pts) For these questions, assume that the initial register values are as follows:
    $t1 = 0x0004, $s0 = 0x01A7, $a1 = 0x6B20, $sp = 0x8034
    Now assume that the following MIPS code is executed:

```
addi   $sp, $sp, -4
sw     $s0, 0($sp)
add    $s0, $zero, $zero
add    $t1, $s0, $a1
```

    After execution, what are the values of $t1, $s0, $a1, and $sp?
    9/27/18 Clarification: provide each value in hexadecimal.

5. Note: For this problem you must connect to **flip.engr.oregonstate.edu** using an SSH client. For help, see the links:
   Windows Instructions
   Mac/Linux Instructions

A properly submitted, correctly operational C file is worth 20 pts.

a. Create a file named **prime_number.c** and insert the starter code below. You will then implement an algorithm (of your choosing) in C to determine whether a provided number is prime.

```
/*
        Short program to determine whether a provided positive number is prime.
*/
int main() {
        // do not change this section
        // you MUST retain the following two variable names:
        int num = 59;
        int is_prime;

        // your code goes in between the following markers
        // <------------->

        // <------------->


        // should return 0 if "num" was not prime
        // should return 1 if "num" was prime
        return is_prime;
}
```

b. Compile your code and **ensure that the algorithm works** by executing the following commands at the terminal prompt:

```
# compile the code
gcc -std=c11 ./prime_number.c
# print the return value and see if it is 1 or 0
a.out; echo $?
```

c. Try changing the value of **num** (and recompiling the code) to make sure that your program properly detects prime numbers.

d. Once your program is working properly, run the following command to generate the x86 assembly code. The output will be stored in a file named prime_number_x86.s

```
# generate x86 assembly code
gcc -fno-asynchronous-unwind-tables -std=c11 -Og -S -o ./prime_number_x86.s ./prime_number.c
```

e. (3 pts) x86 assembly uses 8 32-bit registers. They are: eax, ebx, ecx, edx, esi, edi, ebp, and esp. Which of these registers are used in the assembly code generated in step (5d)?

f. Run the following command to generate MIPS assembly code. The output will stored in a file named prime_number_mips.s

```
# generate MIPS assembly code
mips64-linux-gnu-gcc -fno-asynchronous-unwind-tables -std=c11 -Og -S -o ./prime_number_mips.s ./prime_number.c
```

g. (4 pts) As noted in the textbook, MIPS has 32 32-bit registers (see figure 2.1 in the textbook). Which of these registers are used in the assembly code generated in step (5f)?