Homework 7 [Due at lecture on Fri, Apr 24]

Primary contact for this homework on Piazza is Sujith Surendran

Important Notes:

- Answers to questions 1 and 2 should be handed to your instructor in hard copy on the due date during the class time.

- Problems 3 and 4 should be submitted electronically to learn@UW. Submission guidelines for Problems 3 and 4:
  
  ○ If you have first name Jane and last name Doe, turn in one zip file Doe_Jane.zip into the dropbox folder named “homework7” by 1:59PM on Friday, April 24.

  ○ This zip file should contain the folder Doe_Jane with files hw7_p3.txt and hw7_p4.txt in the folder. These .txt files should contain the assembly code for problems 3 and 4 respectively.

  ○ Since your code will be tested automatically, it is important to stick to this naming convention, otherwise you will lose credit, even if your code is working correctly.

- The programs which you write should always start at address x3000 and end with a HALT instruction (HALT).

- You can submit your code for problem 3 and 4 (ie, the .zip file) as many times as you want, until 1:59 PM on Friday, April 24, 2015. We will consider your latest submission for grading.
Consider the following assembly program:

```assembly
.ORIG x3000
LD R2, INPUT_NUM
LD R3, BITS
AND R0, R0, #0
ADD R1, R0, #1
NEXT AND R4, R2, R1
BRnp SKIP
ADD R0, R0, #1
SKIP ADD R1, R1, R1
ADD R3, R3, #–1
BRzp NEXT
ST R0, RESULT
HALT

INPUT_NUM .FILL x3210
BITS .FILL xF
RESULT .BLKW #1

.END
```

a) (2 Points) Run the program on PennSim and give a brief explanation of what the program does. (i.e., Specify how the value at RESULT relates to value at the INPUT_NUM when the program reaches HALT.)

b) (1 Point) How many times does the instruction at label NEXT execute?

c) (1 Point) What value will be contained in R0 after the execution of the program?
Problem 2 (6 Points)

Consider the following assembly code:

```
.ORIG x4000
LEA R3, INPUT
LD R1, SIZE
ADD R3, R3, R1
LOOP
  LDR R0, R3, 0
  ADD R3, R3, -1
  ADD R1, R1, -1
  BRp LOOP
HALT
INPUT .STRINGZ "Problem_2"
STRING .BLKW 5
SIZE .FILL x12
.END
```

a) (4 Points) In the first pass, the assembler creates the symbol table. Fill in the symbol table created by the assembler for this program (You may extend this table to any number of rows)

<table>
<thead>
<tr>
<th>Symbol Name / Label</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

b) (2 Points) In the second pass, the assembler creates a binary (.obj) version of the program, using the entries from the symbol table. Write the binary code generated for the first two instructions (LEA and LD)
Problem 3 (12 Points)

Assume that you are appointed as a TA for this CS/ECE 252 course in Spring 2016, and assume that 20 students have enrolled in this course. After grading all their exams, you finally decided to use LC-3 for calculating the performance of the class. For doing so, assume you have stored the total marks of each of the students in consecutive memory locations starting at 0x4000.

Write a program in LC-3, which starts at memory location 0x3000, to do the following:

a) (2 Points) Calculate the maximum marks obtained by student and store this value in the memory location corresponding to label “MAX_MARKS” (see template hw7_p3.txt).

b) (2 Points) Calculate the minimum marks obtained by a student and store this value in memory location corresponding to label “MIN_MARKS”.

c) (5 Points) Calculate the average marks of the class and store this value in memory location corresponding to label “AVG_MARKS”. If this turns out to be a fraction, round it off to the next higher integer. For example, if the average turns out to be 30.02, the average should be rounded off to 31.

d) (3 Points) Calculate the number of students who have scored below class average (which has been rounded off) and store this value in memory location corresponding to label “BELOW_AVG”.

Note:
- You MUST use this template for writing your code: hw7_p3.txt.
- You can assume that addition of marks of all the students will not generate overflow.
- You may use the script file (hw7p3_script.txt) for testing your code. The script has some default values for marks for each student. However, please note that we will be finally testing your code with other values for marks (not with the default ones).
  - Before you use the script, make sure that your code is getting compiled successfully.
  - Then you can open Pennsim (if it is closed) and type “script hw7p4_script.txt” in the command line of Pennsim. This script will reset the system, load your code and OS and then tests your code with some default values of marks. If any of the tests fail, you will get a FALSE output on the screen. If all tests are successful, you will see the following 4 true statements (for the default values):
    ■ TRUE (check MAX_MARKS 100)
    ■ TRUE (check MIN_MARKS 70)
    ■ TRUE (check AVG_MARKS 84)
    ■ TRUE (check BELOW_AVG 10)
Problem 4 (8 Points)

Write a program which reads a string of lowercase alphabets starting at label "INPUT". It then checks if the string is a palindrome or not. (A palindrome is a word or phrase that reads the same forward or reversed. For example, “kayak” and “racecar” are palindromes.) If the string is a palindrome, the program should store a value 1 in the memory location corresponding to label “PALINDROME”. Else, it should store a value 0 into this memory location.

Note:

- The string is always terminated by a NULL character (ASCII value 0x0). If the string is empty (the character at label INPUT is the NULL character), the value at PALINDROME should be 0.

- Use this template for writing your code hw7_p4.txt: