CS224 - Fall 2016 - Lab #2 (Version 2: Oct. 18, 2:52 am)

Creating and Running Simple MIPS Assembly Language Programs

Dates: Section 3, Wednesday, 19 October, 08:40-12:30
          Section 1, Wednesday, 19 October 13:40-17:30
          Section 4, Thursday, 20 October 13:40-17:30
          Section 2, Friday, 21 October 13:40-17:30

Purpose: More experience with MIPS programming and subprograms.

ALL GROUPS: Dear students as announced in the classroom please bring and drop your preliminary work into the box provided in front of the lab by 5:00 pm Tuesday October 18.

At the end of the lab before submiting your code for MOSS similarity testing please READ the instructions given here

Students who do not follow the instructions will get 0.

Submit your MIPS codes for similarity testing to the Unilica > Assignment specific for your section. You will upload **one file**: name_surname_MIPS.txt created in the relevant parts. Be sure that the file contains exactly and only the codes which are specifically detailed below. Check the specifications! *Even if you didn't finish, or didn't get the MIPS codes working, you must submit your code to the Unilica Assignment for similarity checking*. Your codes will be compared against all the other codes in the class, by the MOSS program, to determine how similar it is (as an indication of plagiarism).

We plan to run your code to make sure that it really works.

You have to show your work to your TA before by 12:00 in the morning lab and by 5:00 in the afternoon lab. Noe that you cannot wait for the last moment to do this. Make sure that you have shown your work to the Lab TA before uploading. If you wait for the last moment and show your work after the deadline time (12 or 5) 30 points will be taken off. If you upload before showing your work to the TA you will receive 0 (zero) for the lab work.

If we suspect that there is cheating we will send the work with the names of the students to the university disciplinary committee.

Part 1. Preliminary Work
     (30 points)

You have to provide a neat presentation prepared by <u>Word</u>. At the top of the paper on left provide the following information and staple all papers. In this part provide the program listings with proper identification (please make sure that this info is there for proper grading of your work, otherwise some points will be taken off).

CS224 / <u>Your Section No.</u>
Fall 2016
Lab No.
Your Name

1. Write separate MIPS functions that

    1. receives a pointer to an asciiz string that contains a decimal number in the form of a string, like "123", and returns its numeric equivalent in register $v0.

    2. receives an integer value in $a0 and returns it binary equivalent in an array whose address is provided in $a1.

    3. receives a pointer to an asciiz string and converts all of the smallcase letters to their uppercase equivalents and returns the number of such conversions done in $v0.

       For unclear parts make proper assumptions and state them in your comments. The same is also   true for the following parts.

Part 2. String Processing
     (70 points)

1. Write a program to count the number of non overlapping repetitions of a character pattern in a character string. For example " the pattern "aa" appears twice in the in the string "aabbaaa". Provide a user interface to read the character pattern and the string.

2. Write a program that translates a character string (only smallcase letters) into another string. Firstly it gets the translation rule in the form of two parallel arrays (two separate

inputs) original and translation code You may assume proper input. As a challenge you may implement a defensive program that protects itself from various possible user inputs; for example, input arrays with separate length, more than one code for a single character, etc.

Original	a	b	c
Code	x	y	z

After these two inputs the program gets the actual input string to be translated and does the translation. For example, using the above information for the following input string

aabbccddx the output becomes xxyyzzddx. The last two characters remain the same since no translation code is defined for d and x.

Part 3. Cleanup

1) Erase all the files that you created.
2) When applicable put back all the hardware, boards, wires, tools, etc where they came from.
3) Clean up your lab desk, to leave it completely clean and ready for the next group who will come.