LSU EE 3755 Homework 5 Due: 22 November 2013

Problem 0: **Complete this problem as soon as possible.** Follow the instructions for *Account Setup* and *Verilog Homework Workflow* on `http://www.ece.lsu.edu/ee3755/proc.html`.

When the account is set up copy the assignment into your class account using the following commands:

```
[ee37551@frost ~]$ cd ~
[ee37551@frost ~]$ cp -r /home/faculty/koppel/pub/ee3755/hw/2013f/hw05 .
[ee37551@frost ~]$ cd hw05
[ee37551@frost hw05]$ ls
hw05.s
```

The first two commands above copy the files, the last two verify that they have been copied by showing a directory listing.

Start Emacs and load the assignment, `hw05.s`, into an Emacs buffer. If the instructions were followed correctly the MIPS assembler comments should appear in red and the text "LSU EE 3755..." should appear **red bold**. There should also be an "EE 3755" entry on the menu bar.

Problem 1: *Whitespace* in a string or text file refers to spaces, tabs, carriage returns, and other invisible characters that affect the positioning of visible characters. In many programming languages the programmers are free to insert whitespace to improve the readability of their programs, for example, to indent code inside of loops. However some users will just be sloppy with their whitespace. Consider an ordinary text string such as `"I am neat."` There is one space between each word and no extra spaces anywhere. In contrast `" I  am sloppy  . "` contains extra space at the beginning and end, more than one space between some words and space between the period and the wye.

As one might imagine it would take considerable effort just to convince a person who is *whitespace-sloppy* that in fact he or she has a problem that needs fixing. Here we will take the easier approach, writing a program to clean up whitespace, it will be called `wscleanup`.

MIPS assembly language routine `wscleanup` is called with three arguments, in registers `a0`, `a1`, and `a2`. Register `a0` holds the address of the string to clean, register `a1` holds the address at which to write the cleaned string, and register `a2` holds the address of what we'll call the no-space-to-the-left table.

The no-space-to-the-left table has 256 1-character entries. If entry number i contains a 1 that means a character with ASCII code i should not be preceded by spaces. The ASCII code for a period (`.`) is 46, and so entry number 46 would be a 1. The ASCII code for `A` is 65, entry 65 would be 0.

Complete `wscleanup`, which can be found in `hw05.s`, so that it writes memory starting at `a1` with a cleaned version of the string starting at `a0` and uses the table at `a2`. In particular:

- The cleaned string should not start with or end with a space.
- The cleaned string should not contain more than one consecutive space anywhere.
- There should not be a space immediately before any character in the no-space-to-the-left table.

Run the code by pressing F9 to verify correct operation. See the file `hw05.s` for more information.