CS154 Computer Architecture
Winter Term 2019

Lab Assignment 01
- Download and install SPIM for the operating system of your computer:
  `http://spimsimulator.sourceforge.net`
- Learn the properties of SPIM and how to use it from the web site.

1. Consider the following MIPS assembly program fragment which swaps values of two locations in the memory.
   - The program assumes two parameters `v` and `k` which are in registers `$a0` and `$a1`.
   ```
   swap:   sll $t1, $a1, 2
           add $t1, $a0, $t1
           lw  $t0, 0($t1)
           lw  $t2, 4($t1)
           sw  $t2, 0($t1)
           sw  $t0, 4($t1)
   ```
   - Add commentary to each line and between the lines of the MIPS assembly program for each instruction or a group of instructions about what they accomplish.
   - Add additional instructions to the beginning of the program that places the integer array `v = [13, 17, 19, 23, 29]` in the memory and places its address `v` into the register `$a0`. Also add an instruction that places the value of `k = 3` into the register `$a1`.
   - Add appropriate directives to your program, and execute it in SPIM. Your Lab report should contain the commented program, the screenshots SPIM windows, and should also explain their roles and states, when your program is running.

2. Turn the above swap program above into a procedure by adding a **return** instruction, which takes the CPU back to the calling routine, such as "`jr  $ra`". Write the sorting C code below in the MIPS assembler, which uses the swap function. Add appropriate directives as above, and execute your code on SPIM with an array `v` of length 16. Select a set of random integers and place them into `v` before running the sorting code.
   ```
   void sort (int v[ ], int n)
   {
   int i, j;
   for (i = 0; i < n; i += 1)
       {
       for (j = i - 1; j >= 0 && v[j] > v[j + 1]; j -= 1)
           swap(v, j);
       }
   }
   ```
   You can refer to and use the assembly program in Section 2.13 of the book.