CS2318 Summer 2108 Prof. Tamir	Assignment 3 Due 06-28-2018	06-25-2018

Submission Instructions:

1. Please submit your work directly in TRACS (using the TRACS editor) or as a text/MS-word/PDF attachment by the due date/time.
2. Please use only zip for compression.
3. Please write your name in the assignment header and as a part of the file name of any file attached
4. It must be your own work – a penalty of at least one grade in your final grade and a report to the Dean of Students will result from sharing work or using other people work.
5. Please do not submit your assignment via email. If you miss the deadline, then please submit it on TRACS and send me an email notification.
6. A penalty of 10% per day will be assessed for late submission. In addition, it will be graded only at the end of the semester.

Assignment Instructions:

Write a pseudo C program (based on chapter 2 in the book) and a C or C++ program then use these programs to develop a MIPS main program and a MIPS assembly function.

The interaction between the main program and the function is **solely through the stack**.

The function stores $ra immediately after being called and restores $ra before returning to main.

The main program reserves a static C like array (index starts from 0) of 10 elements and initializes it. The maximum should be in locations 4 and 5 (same number) and the minimum in location 7. Locations 0, 1, 8, and 9 should contain the same value. Furthermore, the main program stores the base address of the array in $t0 and the array size in $s0 and has to use these registers after calling the function (you do not have to code the register usage).

The function receives the starting address of an array of 32-bit integers and the number of elements in the array from the main program, finds the maximum integer, the minimum integer within the array, and the average value of the elements in the array and returns the maximum, minimum, average, and the locations of the minimum and maximum values in the array to the main program. Further assume that the function uses $t0 as a pointer to array elements. Additionally, the function stores the index used to "navigate" through the array in $s0. Hence, both $t0 and $s0 are changed by the function.

After calling the function, and receiving the results, the main program prints these results to the screen and stops.

Your deliverables should be the C / Pseudo C programs as well as the assembly code of the MIPS program/function and a screen shot of your run. Use Google to find out how to print a screenshot from your system.