CS2323 Computer Architecture (Spring 2020)

Assignment 2: MIPS Assembly Programming

Marks: 50
This assignment must be done individually.
Start Date: 30-10-2020
Due Date: 11-11-2020, 11:55 pm

NOTE: 10% is deducted for each late day (including weekend) after the assignment due date.

1. Problem Description

Write C and MIPS assembly code to do the following (Breakup: 75% for MIPS code & 25% for C code):

- Implement the Insert sort algorithm (for integers) [**30 marks**]

Generate N numbers that are in geometric progression: $a, ar, ar^2, ar^3 \ldots ar^{(N-1)}$. Your program should prompt the user to input the number of elements *N*, scale factor *a*, and common ratio *r*. You need to consider three cases for sorting the generated GP sequence: [**20 marks**]

(a) Sequence is in ascending order
(b) Sequence is in decreasing order
(c) Sequence is in random order

Print the total number of comparison operations that the sorting algorithm takes for the above three types of input sequence. In specific, every time you compare two numbers while sorting, you should increment the total number of comparisons by one. As output:

1. **Print each sequence**
2. **The corresponding total number of comparisons.**

2. What to Submit

You should submit "summary.txt/pdf" and the source codes (one '.c' file and one '.asm/s' file) as a .zip file with name <rollnumber>_assgn2 through MS teams. Both .c and .asm/s files should contain insertion sort logic, generate 'N' numbers as per geometric progression, and code for three cases for sorting the generated GP sequence. In short, implement the same logic in two different languages, 'c' and 'assembly'. The "summary.txt" file should contain the count of total number of comparisons, as shown below. Each row should have different values for 'N', 'a' and 'r'. For 'N' pick any integer from 1 to 10. For 'a', and 'r' pick any integer values from 1 to 5.

	Insertion Sort – C program		
	Ascending	Descending	Random
N = a, a=x1, r=y1			
N = b, a=x2, r=y2			
N = c, a=x3, r=y3			

	Insertion Sort – MIPS assembly program		
	Ascending	Descending	Random
N = a, a=x1, r=y1			
N = b, a=x2, r=y2			
N = c, a=x3, r=y3			

Note:

1. Double check that you submit the correct code files. **NO** change requests to the code are considered after the evaluation. Penalty applies based on the last code submission time. Don't give away marks for such silly mistakes.
2. Student queries received within 24hrs would be considered for further evaluation. Queries received by the instructor and TAs after 24hrs will be ignored by default, and don't expect a response.
3. Please put the software name you used if it is other than MARS

3. MIPS Simulator and Resources

- SPIM: http://spimsimulator.sourceforge.net/
  - Programming the QtSpim
- http://courses.missouristate.edu/kenvollmar/mars/