Ho Chi Minh City University of Technology
Faculty of Computer Science and Engineering

Computer Architecture

Assignment - Spring Semester 2019-2020

Notes:

- Students are requested to submit the MIPS program(s)/source code (.asm files) and a report (in pdf) to the elearning no later than 6-June-2020. All files need to be compressed into one .zip file before submitting. Assignments must be done individually.
- Students have to demonstrate program(s) on MARS MIPS. Time and location will be announced later. Students not show up during the demonstration time will get 0 for assignments.
- **Similarity less than 20% in MIPS code is allowed. In other words, you will get 0 if your answers are similar to an another student more than 20%.** We will use the Stanford MOSS system to check the similarity (https://theory.stanford.edu/~aiken/moss/).
- The report should not contains code. Students should present the algorithms as well as the idea in your implementation.

Question 1. Given the following MIPS declaration in the data section of a MIPS program

```
.data
 nums .word <an integer number>
 elems .word <array elements>
```

Where `<an integer number>` will store the number of elements in the array `elems`. `elems` is an integer array whose size is equal to value `<an integer number>`. You are required to choose those values when developing and testing your program.

1. Write a MIPS program that sort the the array `elems` in **descending order** using the **quick sort** algorithm. (*3 points*)
2. Calculate the execution time of your program if one instruction requires 1 ns for processing. (*1 point*)

Question 2. Given the following MIPS declaration in the data section of a MIPS program

```
.data
 nums .word <an integer number>
 elems .word <array elements>
```

Where `<an integer number>` will store the number of elements in the array `elems`. `elems` is an integer array whose size is equal to value `<an integer number>`. You are required to choose those values when developing and testing your program. Write a MIPS program that:

1. sort the the array `elems` in **ascending order** at first; (*3 points*)
2. allow users to input an integer number. The program reports position(s) of the number if it exists in the array. Binary search mus be used in this step. (*3 points*)

——————the end——————