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

Computer Architecture Lab

Assignment - Spring Semester, AY 2021-2022

Notes:

- Students are requested to submit the MIPS program(s)/source code (.asm files) to the BK Elearning system no later than the last lab session of your group. Assignments must be done individually.
- Students have to demonstrate program(s) on MARS MIPS during the last lab session. Students who do not show up during the demonstration time will get 0 for assignments.
- **Similarity less than 30% in MIPS code is allowed. In other words, you will get 0 if your answers are similar to another student's more than 30%.** We will use the Stanford MOSS system to check the similarity (https://theory.stanford.edu/~aiken/moss/).
- The report should not contain code. Instead, students should present the algorithms as well as the idea in your implementation.

Topic: Please design and **write MIPS assembly language** for implementing a text-based 5x5 board Tic-Tac-Toe game for two players with following requirements.

1. During the first turn of both players, they are not allowed to choose the central point (row 3 & column 3).
2. Any player who has 3 points in a row, column or diagonal will be the winner.
3. Players can undo 1 move before the opponent plays.

Rubric for evaluation

1. (*2 points*) **Friendly interface**
   - Students can design and implement an amicable user interface so that players can play easily without any confusion (2 points);
   - Students can design and implement a friendly user interface; however, players face some difficulty when playing the game (1.5 points);
   - Students can design and implement a user interface, but it is not friendly, or players need to do several steps for one move (1 point);
   - Student can design and implement a user interface, but it fails to allow playing (0.5 points)
2. (*5 points*) **Application implementation**
   - Students can implement an excellent application without any errors found (4.0 - 5 points);
   - Students can implement a good application with some minor errors, but players do not need to restart the application to continue (3.0 - 4.0 points);
   - Students can implement the application with some errors that prevent players from playing the game (2.0 - 3.0 point);
   - Students cannot implement the application so that players can play/run (0 - 2.0 points)
3. (*3 points*) **Report**
   - Students write such an excellent report that others can understand without any difficulty (3 points);

- Students write a good report but quite simple or lack of information to understand (2.0 - 3.5 points);
- Students write a report with a lot of code embedded without any explanation (1.0 - 2.0 points);
- Students write a simple report with most of the code attached (0.5 points)

——————the end——————