

Computer Organization Quiz 2 – chap 2.

4/1 15:00 ~ 15:10

True or False (10 pts / problem)

- F 1. MIPS adapts the instructions of "add a1, a1, a2; add a1, a1, a3" instead of an instruction of "add a1, a1, a2, a3" to realize the addition $a1+a2+a3$. This follows the design principle of "smaller is faster".
- T 2. The memory used by 32-bit MIPS CPU is byte-addressed, which means that every 8 bits are grouped as a byte and are assigned with an address. A word has 4 bytes that are addressed with 4 continuous addresses. Thus the address of each word is a multiple of four.
- F 3. In MIPS, the most significant byte of a word is stored in the most address of a word, called as big-Endian.
- F 4. If we want to increase the number of registers in CPU, it is easy to be realized nowadays and there is no other side-effect since current semiconductor technology offers us promising techniques to accommodate much more transistors in a chip.
- F 5. Basic block is a fundamental unit recognized and optimized by compilers. For the purpose of programming flexibility, we can allow a label or branch instruction in a basic block, i.e., in some location of a block except at beginning or end.
- F 6. CPU states are referred to as the contents of registers in a CPU. As proceeding the execution in a procedure call, we need to save the contents of all registers to avoid overwriting the contents of registers in the caller.
- F 7. I-format instructions like lw and sw have one field to store a 16-bit constant. For instructions lw and sw, since they access a word of data and the address of each word must be a multiple of 4, we can ignore the two least significant bits of offset constant in the instruction.
- F 8. As we implement a software program in several files and a compiler successfully compiles the files without any error message, then the subsequent linker also can complete the linking operation ^{without} ~~with~~ any error message.
- T 9. Two compilers compile a program. The executable program with more instruction count is not necessarily to have worse performance than the executable program with less instruction since the program with more instructions may have less CPI.
- T 10. Although Java/JIT already has significant performance improvement, it cannot have similar performance in all kinds of applications compared to C since different algorithms may utilize different kinds of operations/instructions, and thus CPI may vary across different algorithms.