IN1006 Systems Architecture (PRD1 A 2022/23)

Started on	Thursday, 15 December 2022, 5:03 PM
State	Finished
Completed on	Thursday, 15 December 2022, 5:10 PM
Time taken	7 mins 4 secs
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

How many superscalar execution units could be used for the MIPS pseudo-code below? Assume no register forwarding.

1. add a, b, c
2. add c, b ,c
3. add f, c ,d
4. add h, e, f
5. add f, h, b

Select one:

- a. Don't know/no answer
- b. 3
- c. 2
- d. 4
- e. 1 ✔
- f. 5

The number of execution units that can be used is limited to the largest sized block of non-dependent instructions. All instructions have some dependency with another.

The correct answer is: 1

Question **2**

Correct

Mark 1.00 out of 1.00

What is the maximum number of instructions that can be executed simultaneously for the MIPS pseudo-code below?

1. add a, b, c
2. sub d, b ,c
3. add g, e ,d
4. add h, e, f
5. sub a, a, b

Select one:

- a. 4
- b. 5
- c. Don't know/no answer
- d. 2
- e. 1
- f. 3 ✔

Instructions 1 and 2 can be executed at the same time (they can share inputs as they have not changed). There is a dependency between 2, 3 (i.e. d). However 3, 4, 5 have no dependences. Therefore the largest size block of non-dependent instructions is 3. Mixing instruction types has no effect.

The correct answer is: 3

Question **3**

Correct

Mark 1.00 out of 1.00

How many superscalar execution units could be used for the MIPS pseudo-code below? Assume no register forwarding.

1. add a, b, c
2. sub d, b ,c
3. add g, e ,d
4. add h, e, f
5. sub a, a, b

Select one:

- a. 5
- b. 4
- c. 3 ✔
- d. 2
- e. 1
- f. Don't know/no answer

The number of execution units that can be used is limited to the largest sized block of non-dependent instructions. Instructions 1 and 2 can be executed at the same time (they can share inputs as they have not changed). There is a dependency between 2, 3 (i.e. d). However 3, 4, 5 have no dependences. Therefore the largest sized block of non-dependent instructions is 3. Mixing instruction types has no effect.

The correct answer is: 3

Question **4**

Correct

Mark 1.00 out of 1.00

Consider the following MIPS code:

1. add a, b, c
2. add b, c ,d
3. add f, e , b
4. add h, k, g
5. add m, k, w

Which of the following compiler optimisations could avoid data hazards in this code assuming a 4 stage pipeline (i.e., 1. fetch instruction; 2. Decode and load data; 3. Execute instruction; 4. Write to memory) with no forwarding?

a. Do not know the answer.
b. Move instruction 4 before instruction 3.
c. Move instruction 5 before instruction 1.
d. Move instructions 4 and 5 before instruction 3. ✔
e. No compiler optimisation is possible.

Your answer is correct.

Instruction 3 depends on instruction 2 (due to b). Instructions 4 and 5 can be moved before it in the program without affecting the overall program outcome. This move would make instruction 3 to load b in the 7th cycle in a pipeline execution (assuming that instruction 1 starts in cycle 1). In the 7th cycle however instruction 2 will have written b to memory (this would happen in cycle 5 of the pipeline). So the data hazard would no longer be present.

The correct answer is:
Move instructions 4 and 5 before instruction 3.

Question **5**

Correct

Mark 1.00 out of 1.00

Which of the following is *not* a design decision made in the MIPS architecture to support pipelining?

Select one:

a. All instructions have the same length and are therefore easier to decode in a pipeline.
b. Operands are aligned hence no single data transfer instruction will require two memory accesses.
c. Instructions have the same format allowing the data to be fetched before the instruction is decoded.
d. Memory operands only appear in loads and stores allowing the execute stage to calculate addresses.
e. Use of deep pipelines to maximise throughput. ✔
f. Don't know/no answer

The instruction pipelines in MIPS implementations are not especially deep, especially by modern standards.

The correct answer is: Use of deep pipelines to maximise throughput.

Question **6**

Correct

Mark 1.00 out of 1.00

How many superscalar execution units could be used for the MIPS pseudo-code below? Assume no register forwarding.

1. add a, b, c
2. add d, b ,c
3. add g, e ,b
4. add h, g, k
5. add f, a, h

Select one:

- a. 5
- b. 1
- c. Don't know/no answer
- d. 4
- e. 3 ✔
- f. 2

Your answer is correct.

The number of execution units that can be used is limited to the largest sized block of non-dependent instructions. In the above block:

- Instructions 1, 2 and 3 have no dependency.
- There is a dependency between 4 and 3 (due to g).
- There is a dependency between 4 and 5 (due to h).

Therefore the largest size block of non-dependent instructions is 3.

The correct answer is: 3

Question **7**

Correct

Mark 1.00 out of 1.00

In what way are compilers usually used to prevent pipeline hazards?

Select one:

- a. Don't know/no answer
- b. Preventing data hazards by releasing the results before they are written to the registers.
- c. A method to avoid control hazards by predicting what way a conditional branch instruction will execute in advance.
- d. Rearranging the instructions to avoid data hazards where possible. ✔
- e. Use of deep pipelines to maximise throughput.
- f. Rearrange instructions such that an instruction that does not affect the decision is executed after the decision filling the bubble that would occur if the pipeline were to stall.

Compliers can be used to rearrange the instructions to avoid data hazards.

The correct answer is: Rearranging the instructions to avoid data hazards where possible.

Question **8**

Correct

Mark 1.00 out of 1.00

Consider the following MIPS code:

1. add d1, b1, c1
2. sub d2, b1 ,c1
3. add d3, b2 ,c2
4. sub d4, b2, c2

Which of the architectures in the Flynn taxonomy would be best suited for executing the above code?

Select one:

- a. SISD or SIMD
- b. Do not know the answer
- c. SIMD or MISD ✔
- d. MIMD or MISD
- e. MIMD or SISD
- f. SISD or SIMD

Your answer is correct.

Instructions 1-2 fit with MISD as they involve different instructions upon the same data. Instructions 3-4 also fit with MISD for the same reason.

Instructions 1, 3 fit with SIMD as they involve the same instruction upon different datasets. Instructions 2-4 also fit with SIMD for the same reason.

Thus, MISD or SIMD would be the best options for the above code.

The correct answer is: SIMD or MISD

Question **9**

Correct

Mark 1.00 out of 1.00

Consider the following MIPS code:

1. add d1, b1, c1
2. add d2, b2 ,c2
3. add d3, b3 ,c3
4. add d4, b4, c4
5. add d5, c5, b5

Which of the four architectures in the Flynn taxonomy would be best suited for executing the above code?

Select one:

- a. MISD
- b. SISD
- c. Do not know the answer
- d. SIMD ✔
- e. MIMD and MISD
- f. MIMD

Your answer is correct.

Instructions 1-5 above are the same (add) and are executed upon different data. Thus, SIMD is the most suitable architecture.

The correct answer is: SIMD

Question **10**

Correct

Mark 1.00 out of 1.00

Which of the following is the best description of a 'data hazard'?

Select one:

- a. Hardware cannot support a combination of instructions in the same clock cycle, e.g. two simultaneous memory accesses.
- b. Loading instructions into pipeline before the result of a decision is known, e.g. loading instructions after a branch
- c. Waiting for the next instruction to be fetched after a branch is taken.
- d. Instruction depends upon the results of a previous instruction still in the pipeline, e.g. compound math expressions. ✔
- e. Don't know/no answer
- f. A gap in the pipeline where some or all of the stages are not processing an instruction.

A data hazard occurs when an instruction depends upon the results of a previous instruction still in the pipeline, e.g. compound math expressions.

The correct answer is: Instruction depends upon the results of a previous instruction still in the pipeline, e.g. compound math expressions.

◄ Quiz 6 _ Weekly Assessed Quiz 2022

Jump to...

Quiz navigation