IN1006 Systems Architecture (PRD1 A 2022/23)

Started on	Thursday, 15 December 2022, 4:03 PM
State	Finished
Completed on	Thursday, 15 December 2022, 4:24 PM
Time taken	21 mins 41 secs
Grade	**5.60** out of 10.00 (**56**%)

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.

```
add a, b, c
add c, b ,c
add f, c ,d
add h, e, f
add f, h, b
```

Select one:

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

Your answer is correct.

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**

Incorrect

Mark -0.10 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. 3
- b. 2 ✖
- c. Don't know/no answer
- d. 5
- e. 1
- f. 4

Your answer is incorrect.

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 **3**

Incorrect

Mark -0.10 out of 1.00

Which of the following describes best the concept of "out-of-order" execution?

- a. Do not know the answer
- b. Instructions are executed in pipelines.
- c. Instructions are loaded first to cache before brought to the CPU for execution.
- d. An instruction and a data queue are used to regulate the execution of instructions by the CPU.
- e. Instructions are executed in two separate CPUs operating in parallel. ✖

Your answer is incorrect.

Instructions are not given to the CPU for execution immediately when they are due. They are placed in a queue and are moved from it to the CPU only when all the data that they need are available. Also the data D produced by an instruction I are placed in a queue and only when the data of all the instructions that were earlier than I are also written in the queue data, D are written to memory.

The correct answer is:
An instruction and a data queue are used to regulate the execution of instructions by the CPU.

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. Move instruction 4 before instruction 3.
b. Move instruction 5 before instruction 1.
c. Do not know the answer.
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 the best description of a 'stall'?

Select one:

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

A stall occurs in response to a control hazard. It involves waiting for the next instruction to be fetched after a branch is taken.

The correct answer is: Waiting for the next instruction to be fetched after a branch is taken.

Question **6**

Correct

Mark 1.00 out of 1.00

Consider the following MIPS code:

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

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. Move instruction 4 before instruction 3. ✔
- b. No optimisations are possible.
- c. Do not know the answer.
- d. Move instruction 5 before instruction 3.
- e. Move instructions 5 before instruction 4.

Your answer is correct.

The data hazards in this code arise due to instruction 3 (it depends on instruction 1).

Moving instruction 4 before instruction 3 will resolve the hazard as following the move instruction 3 will read data in cycle 5 and by this cycle instruction 1 will have stored its output.

The correct answer is:
Move instruction 4 before instruction 3.

Question **7**

Incorrect

Mark -0.10 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 n, f, b

Select one:

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

Your answer is incorrect.

The number of execution units that can be used is limited to the largest sized block of non-dependent instructions. In the above program instructions 1, 2, 4 and 5 can be executed in parallel as they do not depend on data that any of them generates. Therefore 4 is the number of execution units.

The correct answer is: 4

Question **8**

Incorrect

Mark -0.10 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, d
2. add d, b ,c
3. add g, e ,b
4. add h, g, k
5. add f, l, h

Select one:

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

Your answer is incorrect.

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 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).
- There is a dependency between 1 and 2 (due to d).

Therefore the largest size block of non-dependent instructions is 2.

The correct answer is: 2

Question **9**

Correct

Mark 1.00 out of 1.00

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

Select one:

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

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.

Question **10**

Correct

Mark 1.00 out of 1.00

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

Select one:

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

A structural hazard occurs when hardware cannot support a combination of instructions in the same clock cycle, e.g. two simultaneous memory accesses.

The correct answer is: Hardware cannot support a combination of instructions in the same clock cycle, e.g. two simultaneous memory accesses.

◄ Quiz 6 _ Weekly Assessed Quiz 2022

Jump to...

Quiz navigation

1 2 3 4 5 6 7 8 9 10

Show one page at a time

Finish review