IN1006 Systems Architecture (PRD1 A 2022/23)

Started on	Thursday, 15 December 2022, 5:37 PM
State	Finished
Completed on	Thursday, 15 December 2022, 5:57 PM
Time taken	20 mins
Grade	**10.00** out of 10.00 (**100**%)

Question **1**

Correct

Mark 1.00 out of 1.00

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

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

Your answer is correct.

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 **2**

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

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 **3**

Correct

Mark 1.00 out of 1.00

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

Select one:

- a. Loading instructions into pipeline which try to access memory at the same stage.
- b. Instruction depends upon the results of a previous instruction still in the pipeline, e.g. compound math expressions.
- c. Hardware cannot support a combination of instructions in the same clock cycle, e.g. two simultaneous memory accesses.
- d. Don't know/no answer
- e. A gap in the pipeline where some or all of the stages are not processing an instruction.
- f. Executing an instruction in the pipeline whose execution depends upon a yet unknown condition check. ✔

Your answer is correct.

A control hazard occurs when an instruction I has been loaded to a pipeline although due to some condition check which is already being executed in the pipeline it turns out that I should not be executed.

The correct answer is: Executing an instruction in the pipeline whose execution depends upon a yet unknown condition check.

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

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

Consider the following MIPS code:

```
1. lw  $t1, 0($t0)
2. lw  $t2, 1($t0)
3. add  $t3, $t1, $t2
4. sw  $t3, 0($t1)
5. lw  $t4, 2($t0)
6. add  $t5, $t1, $t4
7. sw  $t5, 1($t1)
8. lw  $t6, 3($t0)
9. add  $t5, $t1, $t6
10. sw  $t5, 2($t1)
```

Which of the following would offer the maximum protection against data hazards in this code without affecting the results of its execution, if a 5-stage pipeline is assumed?

a. Move instructions 3 and 6 after instruction 10.

b. Move instruction 5 and instruction 8 before instruction 3. ✔

c. Move instructions 3 and 4 after instruction 5.

d. Move instruction 5 and instruction 7 before instruction 3.

e. Move instruction 5 before instruction 3.

Your answer is correct.

Data hazards in this code extract arise due to loading data immediately before the computations that use them as will make computations stall whilst waiting loading instructions to complete their execution.

Such hazards can be removed by moving instruction 5 before instruction 3 (this will avoid the hazard between instructions 2 and 3 and the hazard between instructions 5 and 6), and instruction 8 before instruction 3 (this will avoid the hazard between instructions 8 and 9).

The correct answer is:
Move instruction 5 and instruction 8 before instruction 3.

Question **6**

Correct

Mark 1.00 out of 1.00

Match the description for the correct MIPS notation for MIPS Pipeline.

Instruction fetch from memory	IF ✔
Instruction decode and register read	ID ✔
Write result back to register	WB ✔
Access memory operand	MEM ✔
Execute operation or calculate address	EX ✔

Your answer is correct.

The correct answer is: Instruction fetch from memory → IF,
Instruction decode and register read
→ ID,
Write result back to register
→ WB,
Access memory operand
→ MEM,
Execute operation or calculate address
→ EX

Question **7**

Correct

Mark 1.00 out of 1.00

Consider the following MIPS code:

1. add a, b, c
2. add c, b ,d
3. add f, c ,a
4. add k, l, m
5. add h, e, g
6. add f, h, 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. No optimisations are possible.
b. Do not know the answer.
c. Move instruction 4 and 5 before instruction 3.
d. Move instruction 4 before instruction 3.
e. Move instruction 4 and 5 before instruction 3 and swap the order of their execution. ✔

Your answer is correct.

The data hazards in this code arise due to instruction 3 (it depends on instructions 1 and 2) and instruction 6 (it depends on instruction 5).

Moving instruction 4 and 5 before instruction 3 and swapping the order of their execution would remove these hazards. This is because following these changes:

Instruction 3 would try to fetch data in the 6th cycle of the pipeline. In this cycle, instruction 2 will have completed the write to memory stage. Also instruction 6 which depends on instruction 5, would try to read data in cycle 7 by which instruction 5 will have written its output.

The correct answer is:
Move instruction 4 and 5 before instruction 3 and swap the order of their execution.

Question **8**

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

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

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

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

◄ 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