EE3450 Homework 3 & 4

總分 280/280

電子郵件地址 *

oliverfeaqr01@gmail.com

0 分，共 0 分

姓名 (Name) *

吳柏澄

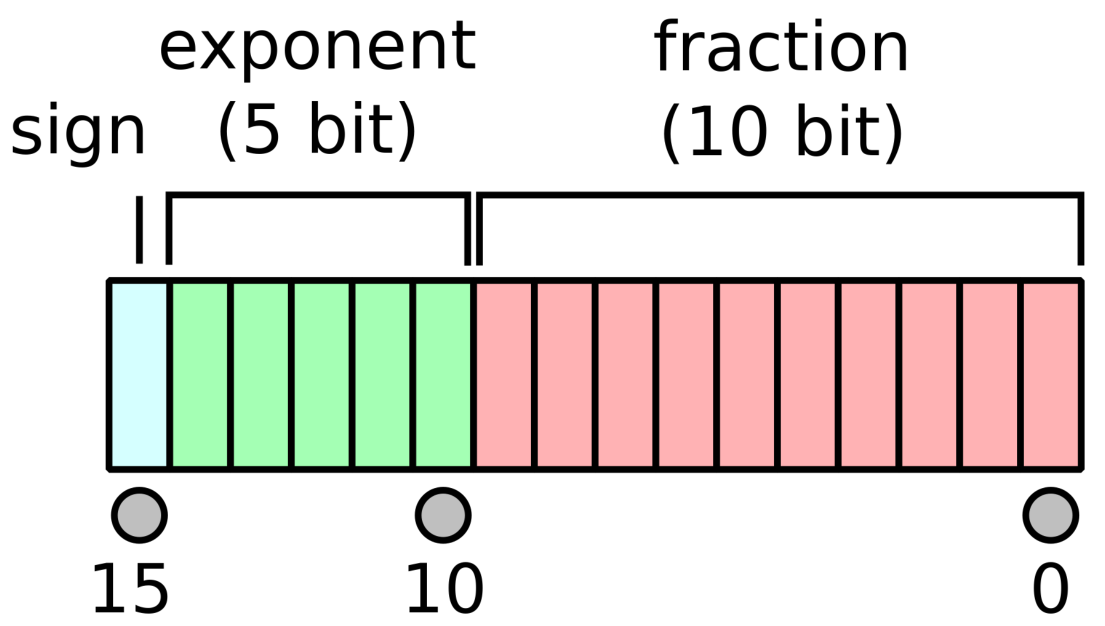
學號 (Student ID) *

106061235

Half float

20 分，共 20 分

Half float is defined as the following figure.



✓ What is the decimal value if 0x3900 is interpreted as a half float? (the bias of half floats is 15) 10/10

0.625 ✓

✓ What is the hex value of a half float whose decimal value is 25.25 10/10

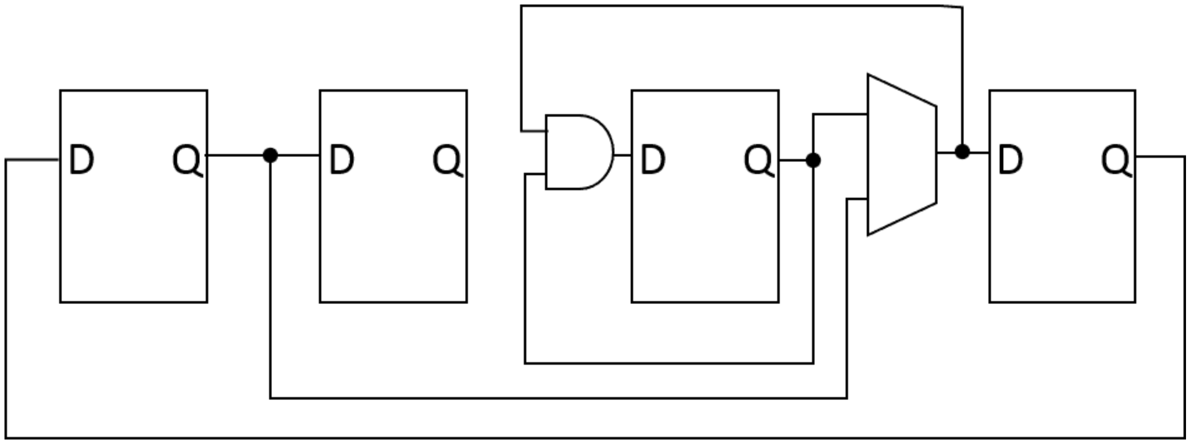
0x4e50 ✓

You can

● continue to the next question

○ submit your answers

Shifter Registers 20 分，共 20 分



✓ If the current bit pattern is 1010 from left to right, please answer a possible bit pattern (in 4-bit binary) of the "next" cycle. 10/10

0111 ✓

✓ If the current bit pattern is 0011, please answer a possible bit pattern (in 4-bit binary) of the "previous" cycle. 10/10

0010 ✓

You can...

● continue to the next question

○ submit your answer

30 分，共 30 分

Consider the following instruction mix: ADD (30%), ADDI (25%), Load (20%), Store (15%), BEQ (10%)

✓ What fraction of all instructions use data memory? (X%, X=?) 10/10

35 ✓

✓ What fraction of all instructions use sign extend? (X%, X=?) 10/10

70 ✓

✓ What fraction of all instructions update the register file? (X%, X=?) 10/10

75 ✓

You can ... *

● continue to the next question

○ submit your answers

60 分，共 60 分

Consider the following program. Assume all registers store 10 initially.
1: ADDI R1, R2, 10
2: ADD R3, R1, R2
3: ADDI R4, R2, 20
4: ADD R5, R3, R1

✓ What is the expected R5 result of this program? 10/10

50 ✓

✓ For a SINGLE-CYCLE CPU, which instruction(s) has data dependency with a previous instruction? 10/10

☐ 1: ADDI R1, R2, 10

☑ 2: ADD R3, R1, R2 ✓

☐ 3: ADDI R4, R2, 20

☑ 4: ADD R5, R3, R1 ✓

✓ For a 5-stage PIPELINED CPU, which instruction(s) has data dependency with a previous instruction? 10/10

☐ 1: ADDI R1, R2, 10

☑ 2: ADD R3, R1, R2 ✓

☐ 3: ADDI R4, R2, 20

☑ 4: ADD R5, R3, R1 ✓

✓ For a SINGLE-CYCLE CPU, how many stall cycles are needed to correctly execute the above program. 10/10

0 ✓

✓ For a 5-stage PIPELINED CPU, how many stall cycles are needed to correctly execute the above program. Assume an ID stage can return the WB results occurring during the same cycle, but NO other forwarding hardware is implemented in the CPU. 10/10

3 ✓

✓ Following the above question, what is the results if this 5-stage CPU executes the program without any stalls? 10/10

30 ✓

You can *

● Continue to the next question

○ Submit your answer

Speeding up a program

80 分，共 80 分

Consider the following program.

```
LOOP:
1:   LD R1, 0(R3)
2:   LD R2, 8(R3)
3:   ADD R4, R1, R2
4:   ADDI R3, R3, 16
5:   BNE R4, R5, LOOP
```

✓ What is the most possible function of the program? 10/10

○ Copying an array to another array

● Searching for a condition that satisfies an equation ✓

○ Counting the number of zeros in an array

○ Finding the minimum value in an array

✓ What is the cycle number a 1-stage pipeline needs to perform the above loop for 100 times until the last branch resolves the condition? 10/10

500 ✓

✓ What is the cycle number a standard 5-stage pipeline needs to perform the loop for 100 times until the last branch resolves the condition. Assume that 1) forwarding is available, 2) no structural hazards, 3) branch is resolved using the EXE stage ALU, 4) no branch prediction. 10/10

800 ✓

✓ Does the 5-stage pipeline complete a program using less cycles than the 1-stage pipeline does? 10/10

○ Yes

● No ✓

✓ What is the speedup of the 5-stage pipeline over the 1-stage pipeline? Assume the clock frequency of the 5-stage pipeline is 4x higher than that of the 1-stage pipeline . 10/10

2.5 ✓

✓ Following the 5-stage questions, what is the cycle number if we exchange the order of the ADD and ADDI instructions? 10/10

700 ✓

✓ What are the advantages of reordering the ADD and ADDI instructions in the above question? 10/10

☐ Reduce data dependencies

☑ Reduce data hazards ✓

☑ Hide the latency of load instructions ✓

☑ Increase the IPC (instruction per cycle) of a pipelined CPU ✓

☐ One should not do such reordering, which changes the results of the program

✓ What are the advantages of reordering the ADDI and BNE instructions in the above question? 10/10

☑ Reduce the bubbles in the pipeline ✓

□ The branch can be resolved 1 cycle earlier

□ The branch can be resolved 1 cycle later

□ Reduce data dependencies

□ One should not do such reordering, which changes the results of the program

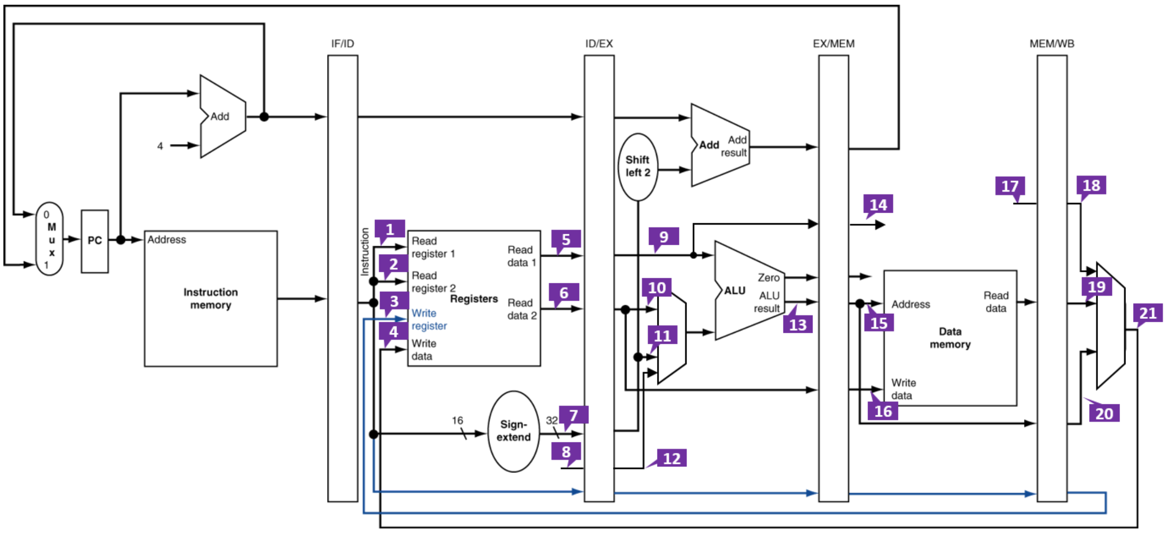
You can

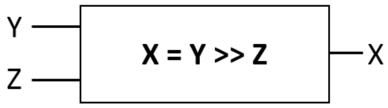
● continue to the next question

○ submit your answers

Pipeline 70 分，共 70 分

We want to add a new instruction:
Shift Rd, Ra, imm(Rb) # Rd= Ra >> (Rb+imm)





✓ To add the instruction (shown in the figure), where should X be connected to? (Please fill in a number betwee 1 and 20). 10/10

17 ✓

✓ Where should Y be connected to? (Please fill in a number between 1 and 20). 10/10

16 ✓

✓ Where should Z be connected to? (Please fill in a number between 1 and 20). 10/10

15 ✓

✓ What is the source of the forwarding path for an added shift instruction immediately followed by a store instruction. (Please fill in a number between 1 and 20). 10/10

18 ✓

✓ What is the destination of the forwarding path for an added shift instruction immediately followed by a store instruction. (Please fill in a number between 1 and 20). 10/10

16 ✓

✓ With the forwarding paths, what is the max number of stall cycles between a load instruction and an added shift instruction? 10/10

○ 0

● 1 ✓

○ 2

○ 3

✓ With the forwarding paths, what is the max number of stall cycles between two consecutive added shift instructions? 10/10

○ 0

● 1 ✓

○ 2

○ 3