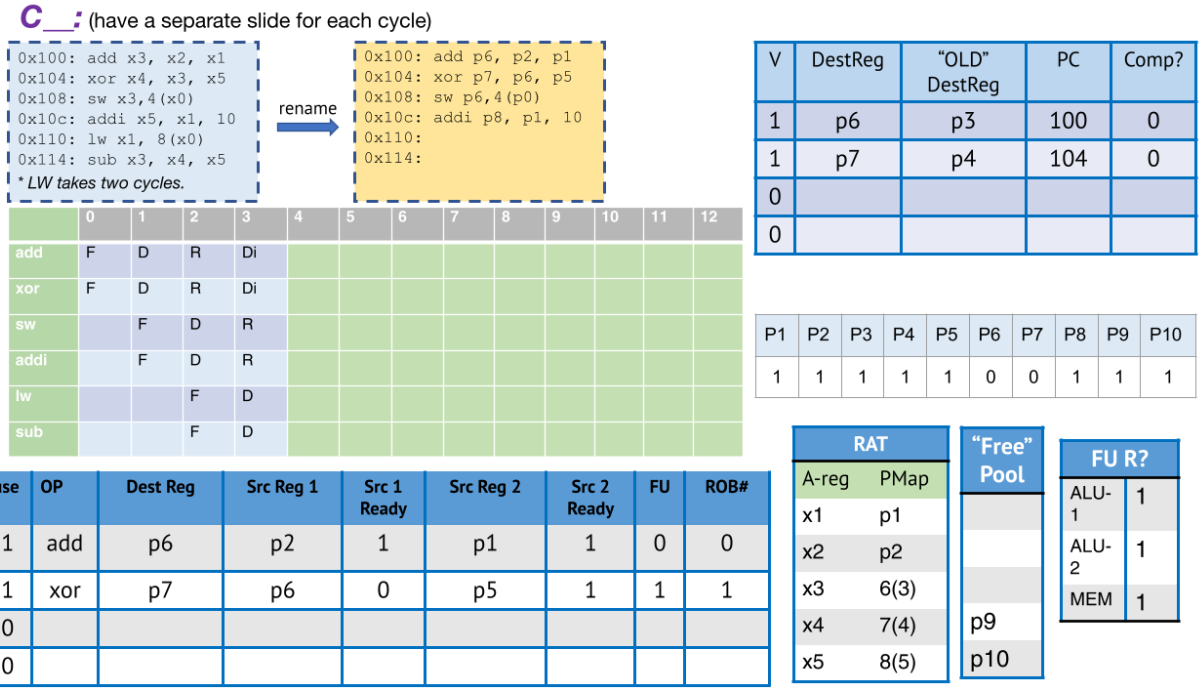
Quiz 3 (Upload it to Gradescope)

Fall 2021 Don't forget to sign in when you are leaving.

Q1. We have the following set of instructions (every instruction takes one cycle to complete except LW):

```
0x100: add x3, x2, x1
0x104: xor x4, x3, x5
0x108: sw x3,4(x0)
0x10c: addi x5, x1, 10
0x110: lw x1, 8(x0)
0x114: sub x3, x4, x5
```

We have a 2-issue out-of-order processor with the following units: (you can access the template by selecting the image and clicking on the link that appears).
(link to ppt).



Start from cycle 4, with Rename, and fill all the tables. In the green table, you should put R, Di (for dispatch), I (for issue, you should not use E), C, and Rt (for retire). If an instruction stays in one stage for multiple cycles (e.g., if the sources are not ready, or if RS is full), repeat the same character for it (e.g., F, D, R, Di, Di, I, C, Rt). Comp in ROB

shows whether the instruction is completed. If an instruction can't be retired, it should stay in C. Also, assume that you can retire a maximum of two instructions per cycle.

Q2. Assume that we have two caches, a fully associative (FA), and a 4-way set-associative. Assume that the addresses are 12 bits. Assume that the FA cache has 8 lines/sets, thus the 4-way set-associative cache should have 2 sets. Answer the following questions:

NOTE: The addresses are CPU addresses (so they need to be shifted).

a. Assuming that each cache line is 16 bytes, what is the block offset, index, and tag size for each cache?

b. Complete the following table for the FA cache. Assume that the cache is filled already except for two lines. ('inv'=invalid and each cell shows the tag). Use (perfect) LRU replacement policy where (7) means the most recently used.

FA LRU	Addresses (coming from the CPU) and tags are shown in HEX. Compute the tag for each Cell/Line (C0, C1, …)								
Address	C0	C1	C2	C3	C4	C5	C6	C7	Hit/Miss
110	11 (6)	inv	15 (2)	28 (3)	12 (4)	18 (5)	22 (0)	79 (1)	M
136	11 (6)	13 (7)	15 (2)	28 (3)	12 (4)	18 (5)	22 (0)	79 (1)	M
121									
141									
220									
111									
C20									
142									
158									
15C									
1A0									
189									
301									

110									
135									

c. Assume that the SA cache uses a **pseudo-LRU** replacement policy. Complete the table for this structure. (if all flag bits are 1, you should reset them). For eviction in each set, start from the left (i.e., W0) and find the first way with flag=0.

SA **PLRU**	Addresses and tags are shown in HEX. Compute the tag for each Way (W0, W1, …).								
	Set 0				Set 1				
Address	W0	W1	W2	W3	W0	W1	W2	W3	Hit/Miss
110	14 (0)	9 (0)	C (0)	11 (0)	3C (0)	8 (1)	A (0)	inv	M
136	14 (0)	9 (0)	C (0)	11 (0)	3C (0)	8 (1)	A (0)	9 (1)	M
121									
141									
220									
111									
C20									
142									
158									
15C									
1A0									
189									
301									
110									
135									

d. What is the miss rate for each case (FA with LRU and SA with PLRU)?

Q3. Provide a short answer for each of the following questions:

A. Do hierarchical page tables improve performance, area overhead, or both? Explain why?
B. How does TLB improve performance?
C. Why “address translation” is needed in a modern processor?