ECS 154B WQ 2019 **Practice Midterm 2**

Topics covered

Book: chapters 4.5 – 4.12, 5.1 – 5.3, and 5.8.
Of course, you'll need to know topics from before the midterm, but most of the focus is on the new material.

Not every topic here is covered in the practice midterm, though most are.
Also check the written homeworks for more problems.

- Pipelining
    - Performance implications
    - Datapath modifications
- Hazards
    - Data hazards
    - Control hazards
    - Branch prediction
- Exceptions
    - Reasoning
    - Types of exceptions
- Instruction-level parallelism (ILP)
    - Definition and finding ILP
    - Static ILP
        - Loop unrolling
        - VLIW and EPIC
        - SIMD
        - Multiple-issue
    - Dynamic ILP
        - Dependencies
        - Out-of-order
        - Register renaming
    - Roofline model
- Memory
    - Characteristics
    - Memory types
    - Memory hierarchy
- Caches
    - Accessing a cache
    - Types of caches
    - Cache parameters
    - Multi-level cache hierarchies
    - Average memory access time (AMAT)

1. Fill in the table with the data hazards in the RISC-V code below. List the instruction address and the register that causes the hazard. Assume the basic 5 stage pipeline with **no forwarding**. Write the addresses of the younger and older instructions. Older instructions are executed before younger instructions (i.e., in cycle 2, addi is younger than lw).

```
0:   lw    x2, 0(x1)
4:   addi  x7, x2, -5
8:   sw    x4, 12(x3)
12:  add   x10, x7, x4
16:  lw    x8, 100(x10)
20:  sub   x10, x9, x8
```

Register number	Younger instruction	Depends on	Older instruction
		➔	
		➔	
		➔	
		➔	
		➔	
		➔	

2. Use the code snippet from the previous problem. Now assume that we are executing that code on an out-of-order processor. Are there any other hazards that we need to worry about? For each hazard, list the hazard type and the instructions involved.

3. Assume we executed the code snippet from the previous two problems on a processor that implemented register renaming. In this processor, why do we not need to worry about the hazards from the previous problem?

4. Intel's Pentium Pro architecture in 1995 had a 256 KB 4-way set-associative L2 cache, with a line size of 32 bytes. Assuming a 32-bit physical address, show how a physical address is laid out for use in the L2 cache. Label each portion of the address, and give the number of bits for each portion.

5. Assume you have a cache that is 4-way set associative and the main design constraints is having the highest hit rate possible (even if it requires more area or power). Would you design this cache with an LRU or random replacement policy?

6. What is a pipeline bubble?

7. Generally, as you increase the number of pipeline stages, does the CPI increase or decrease? Why?

8. What are the three types of hazards? Describe each and describe a technique to reduce their performance impact.

9. Why do we care whether a memory device is volatile? Why would that matter?

10. Name one method for exposing static ILP, and name one for exposing dynamic ILP. Which one is better at exposing more ILP?

11. Explain why predicting all branches as taken is better than predicting all branches as not taken. Writing down a code snippet may help you answer this question.

12. What are the two pieces of information that a branch predictor predicts?

13. Assume the following timings in the table below.

Fetch	Decode	Execute	Memory	Writeback
200 ps	120 ps	250 ps	200 ps	180 ps

All answers must have correct units! (GHz = 10^9 Hz, THz = 10^{12} Hz, ps = 10^{-12} s, ns = 10^{-9} s)

a. What is the cycle time for a single cycle processor, in picoseconds?

b. For a pipelined processor, what is the cycle time assuming each stage is one cycle, in picoseconds?

c. If you were the manager for the next generation of this processor, which stage would you ask your engineers to try to optimize if your goal was to decrease the cycle time?

d. Assume the average CPI with a pipelined processor is 2.4. What is the speedup of the pipelined processor compared to the single cycle processor?

14. On an exception, what two pieces of information must the processor save? What do you do with the pipelined when there is an exception?

15. Can instruction-level parallelism allow the CPI for a processor to be less than 1? Why or why not?

16. List three techniques to decrease cache miss ratio. For each technique, describe its effect on the latency of the cache.

17. Answer the following questions. Express all answers as a power of 2 (e.g., 2^{10} Bytes).

    An address is split into the following parts to access a cache:

```
0000  0000  0000  0000  0000  0000  0000  0000  0000
|___________________________________|___________________|___________|
              Tag (19 bits)               Index (10 bits)   Block offset (7-bits)
```

   a. What is the maximum amount of memory you can address with an address of this size?

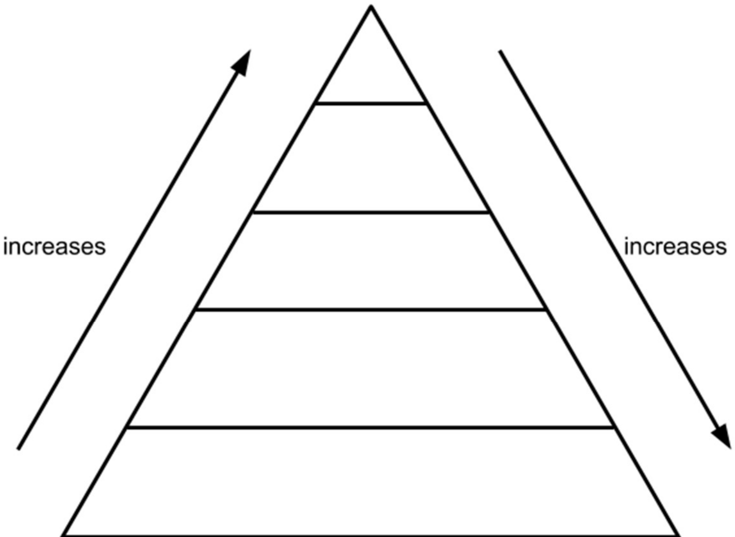
   b. What is the block size of the cache?

   c. How many sets does the cache have?

   d. Assume the cache is 1 MB (2^{20} bytes). What is the associativity?

   e. Ignore part d. What would the cache size be if it was 2-way set associative?

18. Using the diagram below, fill in each step of the memory hierarchy appropriately with the following memory types: DRAM, the cloud, disk memory, SRAM, flash memory, and CPU cache. For each of the arrows, name something that increases as you move in that direction along the hierarchy.



19. Consider the following 512-byte cache that has 16 bytes blocks, 32 entries, and is 2-way set associative. Fill in the table below with the tags for the following address stream. Only write the tag part of the address into the table! Assume the replacement policy is LRU. When something is evicted, mark through the old tag and write the new tag next to it.

    For each access, mark if it is a hit or a miss.

    ```
    0x408ac
    0x408bc
    0x508bc
    0x608bc
    0x00000
    0x408a0
    0x00010
    0x608b4
    ```

Index		
0		

20. Increasing the cache block size for the cache in the previous problem from 16 bytes to 64 bytes will improve application performance, only if it exhibits what kind of locality?

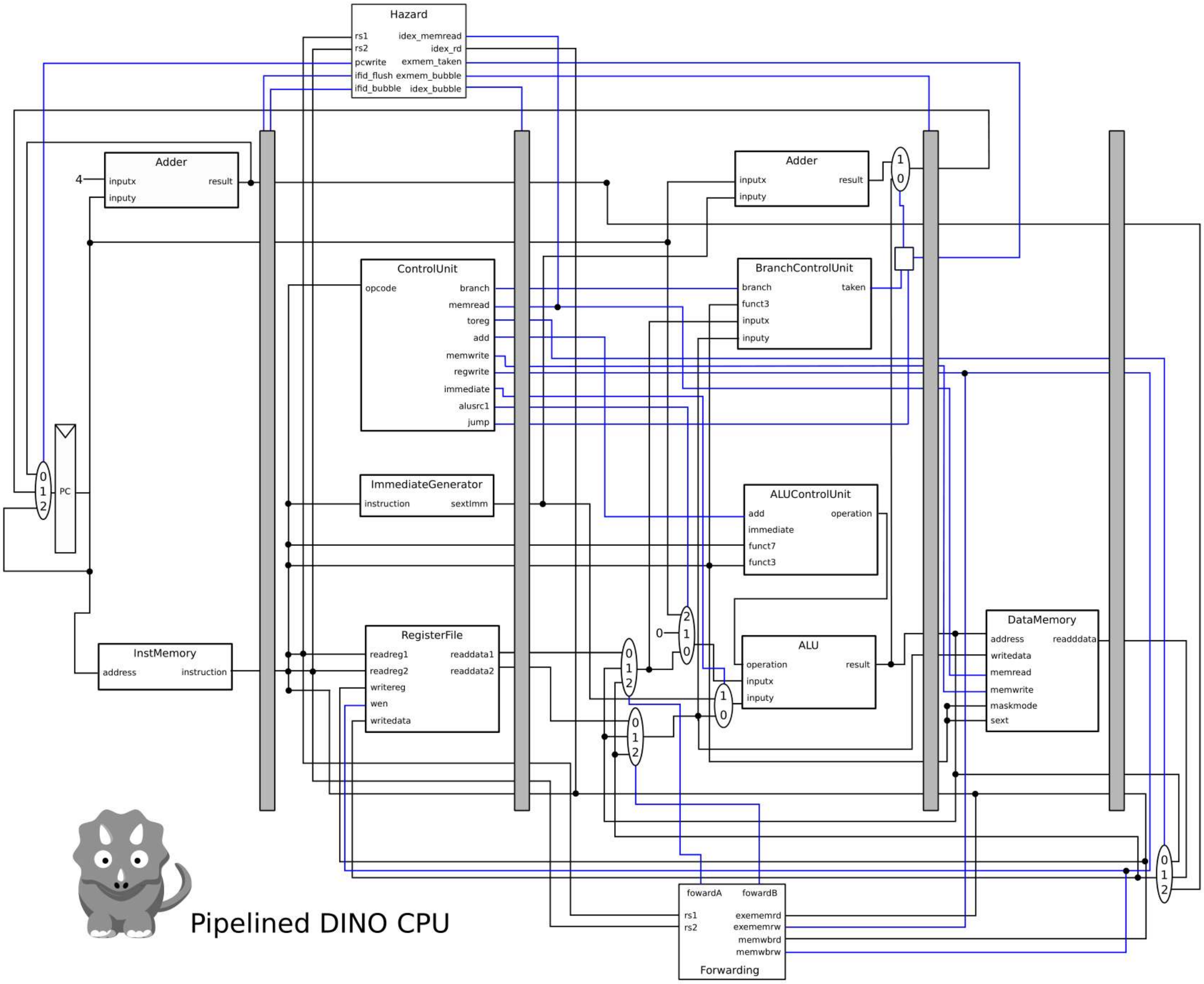
21. Given an L1 cache with a hit time of 2 cycles and a DRAM access latency of 100 cycle, what is the AMAT for a program with a hit ratio of 0.9?

22. What if the above system added an L2 cache with a hit time of 10 cycles and a hit ratio of 0.8?

23. Let's say that the new social networking site for dogs, Püdle, takes 0.5 seconds to render a webpage. That's too long (dogs have a short attention span). Therefore, the Püdle developers instituted a cache of their webpages using memcached. If a webpage is found in the cache, the time to serve the page is only 0.05 seconds. What is the required hit ratio for the cache to make the average wait time for a webpage less than 0.1 seconds?

24. What is the difference between an SRAM cache and an SRAM scratchpad? When would you design a system with a cache, and when would you design a system with a scratchpad?

Use the pipelined DINO CPU diagram below to answer the following questions.



25. When would we select the 1 input on the MUX in the IF stage?

26. How do we bubble the ID/EX pipeline register?

27. Fill in the per-cycle pipeline diagram table for the following code. Show all cycles where forwarding occurs and the stages between which the data is forwarded. Assume there is no branch prediction and branches are resolved in the decode stage. The branch is resolved not taken in the decode stage.

    Use a pencil for this question!

```
0:  add  x3, x7, x11
4:  sub  x2, x4, x5
8:  add  x6, x3, x2
12: lw   x8, 20(x6)
16: beq  x8, x0, 4
20: add  x4, x7, x9
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

28. Can we move instruction 20 up between instructions 12 and 16 to potentially remove a stall? Why or why not?