12 BONUS: Cache Reverse Engineering [70 points]

You are trying to reverse-engineer the cache associativity in a newly-released system. You already know that the cache has a FIFO replacement policy and 8 blocks with a block size of 4 B. Starting with an empty cache, an application accesses five byte addresses in the following order

$$2 \rightarrow 9 \rightarrow 16 \rightarrow 25 \rightarrow 33$$

Assume you can access three addresses after the above sequence and observe the cache hit rate across these three accesses.

1. [30 points] Which three addresses should you access in order to identify the set-associativity of the cache (1-, 2-, 4- or 8-way)? There may be multiple solutions; ***please give the lowest possible addresses that can enable the identification of the set-associativity***. Please explain every step in detail to get full points.

0 → 8 → 16
Explanation. There are four possible set/way configurations, shown below. Each configuration shows the cache state after the five initial accesses. Rows and columns represent sets and ways, respectively, and the byte address accessed is shown for each occupied set:

(a) **(8 sets, 1 way)**

33
-
9
-
16
-
25
-

(b) **(4 sets, 2 ways)**

33	16
-	-
9	25
-	-

(c) **(2 sets, 4 ways)**

33	9	16	25
–	–	–	–

(d) **(1 set, 8 ways)**

2	9	16	25	33	–	–	–

At this point, all four cache associativities have 100% miss rate since they started cold. In order to differentiate the four cases with *just three* more accesses, we need to induce *different* hit/miss counts in each of the four types of cache associativities. The only way this is possible is if one cache type experiences three hits, another experiences three misses, the third one has one hit and two misses, and the last one has two hits and one miss.
Only two solutions exist to produce this case. In the two solutions, any address in each of the address ranges below can be accessed to reverse-engineer the cache associativity.

- **(0-3)→(16-19)→(32-35)**
- **(0-3)→(8-11)→(16-19)**

Choosing the lowest possible addresses, the correct solution is **0 → 8 → 16**

2. [20 points] What is the associativity of the cache if the cache hit rate observed over the 3 extra addresses accessed in Part (1) were:

Hit rate	Associativity
0	
1/3	
2/3	
1	

Based on the solution to Part (1), these are the cache associativities corresponding to different hit rates.

Hit rate	Associativity
0	4-way, 2 sets
1/3	2-way, 4 sets
2/3	1-way, 8 sets
1	8-way, 1 set

3. [20 points] When you accessed the three addresses you determined in Part (1), you observed a 100% hit rate across these three accesses. Now, your friend asks you to access four more addresses in the following order:

$$32 \rightarrow 0 \rightarrow 8 \rightarrow 28$$

Which of the above four addresses would result in a cache miss?

28.
The cache associativity which provides 100% cache hit for the three extra accesses in Part (1) is **8-way and 1 set**. The three addresses 32, 0 and 8 are already available in the cache before the four new accesses requested by your friend. 28 will result in a miss and will be added to the cache.