

Lab 12 Cybersecurity: Postlab Quiz

⚠ This is a preview of the published version of the quiz

Started: Oct 5 at 3:38pm

Quiz Instructions

Question 1

1 pts

How does the **HashMap Cache** improve upon the **LinkedKVStore Cache**?

- Constant-time lookup
- Bounded size
- Better availability

Question 2

1 pts

How does the **Capped Hashmap Cache** improve upon the **Hashmap Cache**?

- Constant-time performance
- Bounded size
- Only caches legitimate requests

Question 3

1 pts

Consider the CacheList. CacheNode's methods moveToFront() and removeLast(). Their runtime complexity is (let n be the number of items in the list).

- moveToFront is $O(n)$ and removeLast is $O(n)$
- moveToFront is $O(1)$ and removeLast is $O(n)$
- moveToFront is $O(n)$ and removeLast is $O(1)$
- moveToFront is $O(1)$ and removeLast is $O(1)$

Question 4**1 pts**

When using average latency as a performance gauge, the **LRU Cache** improved upon the **Capped Hashmap Cache** (which lacked a suitable eviction policy) by a factor of

- 2:1
- 5:1
- 10:1
- 15:1
- 30:1

Question 5**1 pts**

Which of the following statements is false?

- If a cache can perform key lookup in constant time, and assuming that the miss penalty is constant as well, then its average latency depends solely on the miss rate
- If a cache of size S can absorb a workload without causing cache misses, then so can a cache of $2 * S$.
- Increasing the cache size will always improve the miss rate (i.e., lower it).

Question 6

1 pts

Consider an **LRU Cache** with capacity 3. When faced with a sequence of requests whose keys are "A", "B", "C", "A", "D", the lookup of key "D" and subsequent insertion of its value will:

- Cause a cache miss and "A" to be evicted
- Cause a cache miss and "B" to be evicted
- Cause a cache hit and "B" to be evicted.
- Cause a cache hit and no eviction

Question 7

0 pts

Do you have any feedback/suggestions on these two labs for future offerings?

HTML Editor

B *I* U A ▾ A ▾ I_x x^2 x_2
 \sqrt{x} 12pt ▾ Paragraph

0 words

Not saved

Submit Quiz