Department of Electrical Engineering and Computer Science

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

6.033 Computer Systems Engineering: Spring 2018

Quiz 2

There are **16 questions** and **10 pages** in this quiz booklet. Answer each question according to the instructions given. You have two hours to answer the questions.

- The questions are organized (roughly) by topic. They are not ordered by difficulty nor by the number of points they are worth.
- **If you find a question ambiguous, write down any assumptions you make.** Be neat and legible.
- Some students will be taking a make-up exam at a later date. **Do not** discuss this quiz with anyone who has not already taken it.
- You may use the back of the pages for scratch work, but **do not** write anything on the back that you want graded. We will not look at the backs of the pages.
- Write your name in the space below. Write your initials at the bottom of each page.

This is an open-book, open-notes, open-laptop quiz, but you may **NOT** use your laptop, or any other device, for communication with any other entity (person or machine).

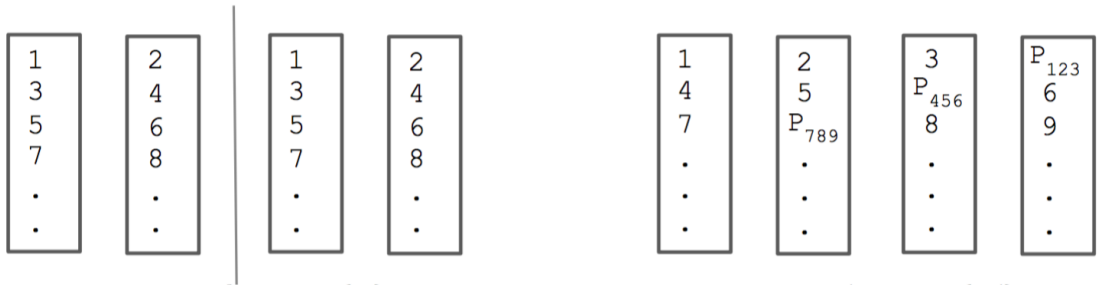
Turn all network devices, including your phone, off.

CIRCLE your recitation section:

10:00	1. Mark	3. Muriel	5. Karen			
11:00	2. Mark	4. Muriel	6. Karen			
12:00	7. Howard	9. Sam				
1:00	8. Howard	10. Sam	11. Mark	13. Peter	15. Tim	17. Adam
2:00	12. Mark	14. Peter	16. Tim	18. Adam		

Name:

1. [8 points]: Consider the two storage organizations shown below.



RAID-1 Storage (R1) RAID-5 Storage (R5)

One uses RAID-1 (R1) and the other uses RAID-5 (R5). Here, each rectangle represents a disk. The numbers represent blocks of data and show how the blocks are striped across different disks. P_{ijk} represents the parity block for data blocks `i`, `j`, and `k`. Assume that disk controllers can detect an erroneous data block while reading it. If a disk fails, then all the blocks on that disk become inaccessible.

A. What is the maximum number of distinct data blocks that can be read concurrently (where each read is processed at the same time on a different disk) by these two schemes?

R1: _____ **R5:** _____

B. How long is the longest sequence of consecutively numbered data blocks that can be read concurrently by these two schemes?

R1: _____ **R5:** _____

C. How many disk blocks are needed to store 6000 blocks of user data in these two systems?

R1: _____ **R5:** _____

D. In case a bad data block is detected while reading, how many disks have to be read to reconstruct the bad block in these two systems (excluding the read to find the bad block in the first place)?

R1: _____ **R5:** _____

Initials:

2. [8 points]: A transaction-based system is using write-ahead-logging. The log is backed by both on-disk cell storage and an in-memory cache. The cache is shared by all transactions. Writes go first to the log, then to the cache. Reads go first to the cache, and then to cell storage if the data is not present in the cache. Assume that all values are initialized to zero in cell storage, and that the cache is large enough to hold all data in question.

Consider the following log snippet. The format for `UPDATE` records is `UPDATE var=<old value> var=<new value>`.

Log Entry	Transaction ID	Record
1	T_1	BEGIN
2	T_1	UPDATE W=0; W=10
3	T_1	UPDATE X=0; X=20
4	T_1	COMMIT
5	T_2	BEGIN
6	T_3	BEGIN
7	T_3	UPDATE Y=0; Y=30
8	T_2	UPDATE Z=0; Z=40
9	T_2	COMMIT

Occasionally, all values in the cache are copied to cell storage. For the purposes of this question, assume that that happens exactly once: immediately before Log Entry 8 is written.

Suppose that the system crashes after Log Entry 9 is written.

A. Immediately before the crash, what are the values in the cache?

W: _____ **X:** _____ **Y:** _____ **Z:** _____

B. Immediately before the crash, what are the values in cell storage?

W: _____ **X:** _____ **Y:** _____ **Z:** _____

After the crash, the system goes through the recovery process. It first does an UNDO phase, followed by a REDO phase.

C. After the UNDO phase, but before the REDO phase, what are the values in cell storage?

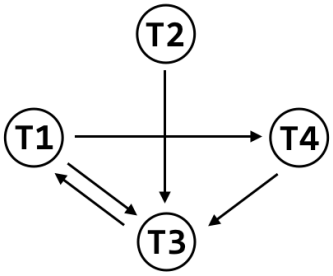
W: _____ **X:** _____ **Y:** _____ **Z:** _____

D. After the REDO phase (i.e., after recovery is complete), what are the values in cell storage?

W: _____ **X:** _____ **Y:** _____ **Z:** _____

Initials:

3. [6 points]: Consider the following conflict graph.



This graph corresponds to the following schedule of operations. Steps 3 and 6 have been left out.

T_1	T_2	T_3	T_4
	1. write(z)		
		2. write(x)	
			4. read(y)
		5. write(y)	
7. write(x)			
		8. read(x)	

A. Determine Steps 3 and 6. For each, specify the transaction (T_1, T_2, T_3, or T_4), the operation (`read` or `write`), and the variable (`x`, `y`, or `z`). If there are multiple options for either step, you need only give one.

Step 3: _____

Step 6: _____

B. Gaby claims that if you begin with the above schedule, but remove a single transaction T_i entirely—i.e., get rid of all of its operations—the resulting schedule will be conflict serializable. What are the possible values of T_i? Circle **all** that apply.

(a) T_1
(b) T_2
(c) T_3
(d) T_4
(e) None of the above (i.e., there is no single transaction that can be removed and result in a conflict-serializable schedule)

Initials:

4. [6 points]: Answer True or False for each of the following questions about GFS and LFS.

A. True / False GFS implements "at least once" semantics for all writes.

B. True / False From the point of view of clients, GFS provides single-copy consistency over its replicas.

C. True / False The commit point of a write in LFS is the point at which the checkpoint region is updated to include the location of the updated inode.

5. [6 points]: Consider a transaction-processing system running three transactions, `T1, T2, and T3`, which read and write three data items, `A`, `B`, and `C`. The system employs two-phase locking at data-item granularity, using reader-writer locks.

Suppose the system runs for some time, and then deadlocks (with none of the three transactions able to make progress). You know that the previous two statements that successfully completed were a write to `A` by `T1` and a read of `B` by `T2`. There have been no commits or aborts since these statements completed. Which of the following could explain the observed behavior? (Circle True or False for each choice.)

A. True / False `T1` is waiting for `T3` to release a lock on `B`; `T2` is waiting for `T1` to release a lock on `A`; `T3` is waiting for `T2` to release a lock on `C`.

B. True / False `T1` is waiting for `T2` to release a lock on `B`; `T2` is waiting for `T1` to release a lock on `A`; `T3` is waiting for `T1` to release a lock on `A`.

C. True / False `T1` is waiting for `T2` to release a lock on `A`; `T2` is waiting for `T1` to release a lock on `A`; `T3` is waiting for `T2` to release a lock on `C`.

6. [4 points]: Answer True or False for each of the following questions about Concurrency Control and Recovery.

A. True / False A system using a Steal/No-Force policy will have better transaction-processing performance than a system using a Force/No-Steal policy, all other things being equal.

B. True / False A system using a Force policy does not need undo-logging.

Initials:

7. [4 points]: Which of the following statements best describes the role of two-phase commit in transaction processing? (Circle the **best** answer.)

(a) It provides all-or-nothing atomicity when a transaction modifies data that is partitioned across multiple machines.

(b) It improves the availability of a multi-node transaction processing system.

(c) It prevents two transactions from concurrently modifying the same data item.

(d) It ensures that multiple nodes will agree on the outcome of some transaction at exactly the same time.

8. [6 points]: Decide which consistency guarantee(s) each of the following implementations satisfy. In each case, circle **all** that apply.

A. A write by a client K for a piece of data `x` is complete once it appears on a majority of replicas. A read by K for `x` is complete once the value has been confirmed by a (possibly different) majority of replicas.

(a) Strong consistency
(b) Eventual consistency
(c) Read-my-writes

B. A write by a client K for a piece of data `x` is complete once it appears on a single replica. A read by K for `x` is complete once the value has been returned by a (possibly different) replica. Replicas perform a background synchronization process to agree on data values.

(a) Strong consistency
(b) Eventual consistency
(c) Read-my-writes

C. A write by a client K for a piece of data `x` is complete once it appears on a single replica. A read by K for `x` is complete once the value has been returned by that same replica. Replicas perform a background synchronization process to agree on data values.

(a) Strong consistency
(b) Eventual consistency
(c) Read-my-writes

Initials:

9. [8 points]: Consider the state of a RAFT system on five servers, S1—S5.

S1	1	1	3	
S2	1	1	2	4
S3	1			
S4	1	1	2	
S5	1	1	2	

Here, each cell represents a log entry. The number in the log entry indicates the term number.

A. Which of the following are consistent with this log? Circle **all** that apply.

(a) S1 was the leader in term 3; S1, S3, and S4 voted for S1 in term 3.

(b) S1 was the leader in term 3; S1, S2, and S4 voted for S1 in term 3.

(c) S3 was the leader in term 1.

(d) S2 was the leader in term 2; S3, S4, and S5 voted for S2 in term 2.

(e) S5 was the leader in term 2; S2, S4, and S5 received the log entry for term 2 before term 3 began.

B. If the system runs for awhile, which of of the following are possible prefixes for the committed state of the log? Circle **all** that apply.

S1	1	1	3	
S2	1	1	3	
S3	1	1	3	
S4	1	1	3	
S5	1	1	3	

(a)

S1	1	1	4	
S2	1	1	4	
S3	1	1	4	
S4	1	1	4	
S5	1	1	4	

(b)

S1	1	1	2	4
S2	1	1	2	4
S3	1	1	2	4
S4	1	1	2	4
S5	1	1	2	4

(c)

10. [6 points]: An attacker is interested in building a data structure to aid in stealing users' passwords. This data structure should allow the attacker to look up any of the k most common passwords, along with a salt, and retrieve the appropriate (salted) hash of the password. For the purposes of this question, we'll call this data structure a rainbow table.[1]

The k passwords are of length at most b bytes, and the hash function in use outputs a d-byte string. The attacker is able to compute the hash of an arbitrary-length string in t time. Each salt is a random bitstring of length ℓ ($\ell < d$).

A. Roughly how much time will it take the attacker to generate a rainbow table for the k most common passwords?

B. Roughly how much storage will the rainbow table consume?

To combat the attacker, the system designer decides to tweak their approach to salted hashes: instead of storing the salt s along with $H(p|s)$, they store s and $H(p|H(s))$. The authentication code changes appropriately (checking that, for an inputted password p', $H(p|H(s)) = H(p'|H(s))$).

C. Which of the following are true? Circle the **best** answer.

(a) This approach is **more** secure against rainbow table attacks, because the attacker won't know that the designer is using $H(s)$ instead of s in the calculation.

(b) This approach is **more** secure against rainbow table attacks, because $d > \ell$ (i.e., because $H(s)$ is longer than s).

(c) This approach is as secure as the original approach.

(d) This approach is **less** secure than the original approach and/or does not allow the designer to properly authenticate valid users in the absence of any attacks.

11. [4 points]: Could DNSSEC help defend against the Torpig botnet? Circle the **best** answer.

(a) DNSSEC would help defend against Torpig because domains would be authenticated.

(b) DNSSEC would help defend against Torpig because DNSSEC encrypts DNS requests.

(c) Both (a) and (b).

(d) None of the above.

[1] An actual rainbow table is a much more complicated structure. We're just interested in a simple lookup table; no fancy compression, etc.

12. [12 points]: Lilika runs a server S, with secret key k. Lilika is interested in using cookies to authenticate users. She is concerned about three types of attacks:

- **Reuse**: Given a copy of a cookie C for username u, an attacker could easily create a cookie C' for a username u' ($u' \neq u$).
- **Extension**: Given a copy of a cookie C for username u, an attacker could easily use C to login as u for an indefinite period of time.
- **Distributions**: Given a copy of a cookie C for username u, an attacker could easily create a cookie C' that would allow the attacker to login to a different server, S' (assuming that S' was using the same cookie scheme as S, but had its own secret key $k' \neq k$).

Decide whether each of the following cookie formats is vulnerable to the above attacks. Below, u refers to a username, exp refers to an expiration date (determined by the server), and H refers to a hash function.

A. $C = \langle u, exp, H(k) \rangle$

(a) **True / False** This format is vulnerable to a reuse attack.
(b) **True / False** This format is vulnerable to an extension attack.
(c) **True / False** This format is vulnerable to a distribution attack.

B. $C = \langle u, H(k|u) \rangle$

(a) **True / False** This format is vulnerable to a reuse attack.
(b) **True / False** This format is vulnerable to an extension attack.
(c) **True / False** This format is vulnerable to a distribution attack.

C. $C = \langle u, exp, H(u) \rangle$

(a) **True / False** This format is vulnerable to a reuse attack.
(b) **True / False** This format is vulnerable to an extension attack.
(c) **True / False** This format is vulnerable to a distribution attack.

D. $C = \langle u, exp, H(k|u|exp) \rangle$

(a) **True / False** This format is vulnerable to a reuse attack.
(b) **True / False** This format is vulnerable to an extension attack.
(c) **True / False** This format is vulnerable to a distribution attack.

13. [4 points]: Answer True or False for each of the following questions about memory corruption attacks.

A. True / False DEP/W$\oplus$X doesn't prevent ROP attacks because ROP doesn't require modifying code.

B. True / False Stack cookies/canaries provide very weak protection to most memory corruption attacks, but are typically deployed anyway.

Initials:

14. [4 points]: Which of the following attacks reveal private information (e.g., private data stored on a server such as passwords or secret keys, private message contents, etc.)? Circle **all** that apply.

(a) Optimistic ACKs

(b) HTTP flooding attacks (where an attacker sends many HTTP requests to a server to get a particularly large file)

(c) DNS reflection attacks

(d) Man-in-the-middle attacks on Diffie Hellman

15. [6 points]: Alice and Bob are setting up a secure channel. An attacker, Eve, is attempting to decrypt their messages. Eve can observe their communications, but **cannot** send any packets (i.e., she cannot create her own packets and send them, nor can she duplicate packets from Alice and/or Bob and send those). Which of the following methods of key-exchange will keep their subsequent communications confidential from Eve? (Circle True if the method keeps their communications confidential, False otherwise.)

A. True / False Alice and Bob decide to use public-key cryptography. They send their public keys over the (insecure) channel before beginning to encrypt their communications.

B. True / False Alice and Bob decide to use symmetric-key cryptography. They each generate their own secret key for communications (they use two separate keys so that messages from Alice to Bob are not valid messages from Bob to Alice). They send their secret keys over the (insecure) channel before beginning to encrypt their communications.

C. True / False Alice and Bob decide to use symmetric-key cryptography, and Diffie-Hellman key exchange. They each pick their random numbers, a and b respectively. Alice sends $g^a \mod p$ and Bob sends $g^b \mod p$ over the (insecure) channel before beginning to encrypt their communications (where p is a large prime number and g is an appropriate generator).

16. [8 points]: Xavier is sending data to a server S using Tor. His data travels through a series of three proxies: P_1, P_2, and P_3.

Assume that Xavier has **not** set up a secure channel with S. Which of the following attacks is Xavier vulnerable to? Circle True or False for each of the following.

A. True / False If the attacker observes both the link between Xavier and P_1, and the link between P_3 and S, they can mount a correlation attack on Xavier's traffic, and infer that Xavier is communicating with S.

B. True / False If the attacker gains control of P_2, and can read all data stored on P_2, as well as observe all packets passing through P_2, they can infer that Xavier is communicating with S.

C. True / False If an attacker observes **only** the link between P_3 and S, they can read the data that Xavier has sent to S.

D. True / False If an attacker observes **only** the link between Xavier and P_1, they can read the data that Xavier is sending to S.

Initials: