CIS 520 - Final Exam – Fall 2013 Name: _____

Due Date: Friday, December 20, 2013, 1:40 pm

1. **(15 pts) Synchronization Problem:** Consider the pseudocode shown below (also available as pthreads code in /pub/CIS520/final/threads.c). After initializing the semaphores, s1 and s2, three threads are created executing runA(), runB(), and runC(), respectively.

```
semaphore s1 = 1;
semaphore s2 = 0;

void *runA() {
 output 'W';
 sem_post(&s2);
 sem_wait(&s1);
 output 'D';
 sem_post(&s1);
 output 'T';
 return;
}
```

```
void *runB() {
 sem_wait(&s2);
 sem_wait(&s1);
 output 'I';
 sem_post(&s2);
 output 'C';
 output 'S';
 sem_post(&s1);
 return
}
```

```
void *runC() {
 sem_wait(&s1);
 output 'L';
 sem_wait(&s2);
 output 'A';
 sem_post(&s1);
 sem_post(&s2);
 return;
}
```

   a. Is it possible for the threads to deadlock? If so, explain briefly how deadlock may occur. If not, describe which condition required for deadlock is not satisfied.

   b. Which of the following outputs are possible? (Circle all that are possible):
      (1) WILDCATS
      (2) WDTLAICS
      (3) WICSLADT
      (4) LWIDCATS
      (5) WLADTICS
      (6) WLAICSDT

   c. Modify the synchronization code, adding at most one more semaphore and/or changing the semaphores' initial count values if necessary and adding or removing semaphore operations, so that the only possible output is 'WILDCATS'. Submit the updated threads.c.

2. **(15 pts) Cryptography Problem:** Using RSA, a public key (e, n) and a private key (d, n) can be constructed from prime numbers, p = 13 and q = 31, by setting n = p*q = 403. Then, if we select e = 7, the encryption key is (7, 403).

   a. Compute the corresponding decryption key and determine how the data m = 66 would be encrypted.

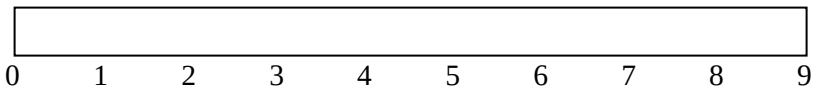
   b. Then, show how the ciphertext c = 326 would be translated back into plaintext m, and how the plaintext value m can be computed efficiently by hand assuming that you are somewhat good at arithmetic.

   c. It is not uncommon for a company to compute the value of a cryptographically secure checksum, using a hash function such as MD5 or SHA1, for a file to be distributed, and then encrypt this value with a private key whose corresponding public key is advertised by the company. What is the purpose of this sequence of operations, and what can the resulting encrypted hash value be used for?

3. **(15 pts) Process Scheduling Problem:** Suppose that the following processes arrive for execution at the times indicated below. Each process will run for the amount of time listed under Run Time. Assume that **non-preemptive** scheduling is used.

Process	Arrival Time	Run Time
P1	0.0	3.0
P2	1.0	5.0
P3	2.0	1.0

a. What is the average turnaround time for this set of processes if the First-Come, First-Served (FCFS) Scheduling Algorithm is used and processes are scheduled non-preemptively? Complete the Gannt Chart to show how the processes would be scheduled.



b. What is the average turnaround time for this set of processes if the Shortest Job First (SJF) Scheduling Algorithm is used and processes are scheduled non-preemptively?

c. Is the schedule generated using SJF optimal for non-preemptive processes? _____. If it is not optimal, generate another non-preemptive schedule that results in a smaller average turnaround time; e.g., draw a Gannt Chart showing an optimal schedule. Otherwise, explain briefly why SJF is always optimal.

d. What is the average response time for this set of processes if the First-Come, First-Served (FCFS) Scheduling Algorithm is used?

e. What is the average response time if SJF is used?

4. (15 pts) File System Problem:

a. Explain why a hard link in Unix cannot span different file systems while a soft link can. (Hint: Think about what meta-data is stored for each type of link.)

b. Explain why in a Unix file system deleting a hard link to a file would require updating the file's inode while deleting a soft link would not.

c. Which of the following conditions would likely represent a serious problem with a file system (Justify answer for each.)

(a) A write to a data block that contains no on-disk inodes pointing at it.
(b) A write to a data block that contains multiple on-disk inodes pointing at it.
(c) A write to a data block that is marked as free in the on-disk bitmap.

5. **(20 pts) Memory Management Problem:** Starting with no pages in memory, consider the following page reference string: 1, 2, 3, 4, 5, 4, 3, 2, 1, 2, 3, 4, 3, 2, 1.

   a. How many page faults would result for the FIFO Page Replacement Algorithm if there are 3 page frames as shown below? _____ Also, complete the table showing where each page will be loaded in memory. What is the hit rate? _____. Hint: the hit rate is the percentage of references resolved without causing a fault.

FIFO	1	2	3	4	5	4	3	2	1	2	3	4	3	2	1
Frame 1	1	1													
Frame 2		2													
Frame 3															
Fault (y/n)	y	y													

   b. How many faults are generated if the Least-Recently-Used (LRU) Page Replacement Algorithm is used instead? _____. Also, complete the table showing where each page will be loaded. What is the hit rate? _____.

LRU	1	2	3	4	5	4	3	2	1	2	3	4	3	2	1
Frame 1	1	1													
Frame 2		2													
Frame 3															
Fault (y/n)	y	y													

   c. How many faults are generated if the Second-Chance (Clock) Page Replacement Algorithm is used? _____. Also, complete the table showing where each page will be loaded. What is the hit rate? _____.

CLOCK	1	2	3	4	5	4	3	2	1	2	3	4	3	2	1
Frame 1	1	1													
Frame 2		2													
Frame 3															
Fault (y/n)	y	y													

   d. Assume that you have a machine with a fixed amount of physical memory and a demand-paged virtual memory system. Is it possible that doubling the page size can reduce the number of page faults? If so, describe how. If not, describe why?

   e. For the same problem, is it possible that halving the page size can reduce the number of page faults? If so, describe how. If not, describe why?

6. (20 pts) Protection and Security Questions:

a. What is the difference between a capability-based system and an access control list-based system?

b. Would having a lock on every file that requires a special software key to open be considered a capability-based system or an access control list-based system? Justify your answer.

c. Is it easier to revoke permissions in a capability-based system or in an access control list-based system? Explain briefly.

d. The Unix login program checks whether or not a user has entered the correct password. If the user is authentic, then the login program executes the user's shell. Explain how Unix can check passwords without storing the user's actual password on the system?

e. Suppose an attacker breaks into a Unix machine, obtains root (superuser) privileges, and manages to keep them for a long period of time (e.g., many months). What might the attacker do to learn users' real passwords, even if they aren't stored on the system?