

Homework 9

Due: never, but think about these before the final exam

1. Tanenbaum, p. 578, Question 2.
2. Tanenbaum, p. 578, Question 3.
3. Tanenbaum, p. 579, Question 14.
4. Tanenbaum, p. 580, Question 25.
5. Tanenbaum, p. 581, Question 34.

The remainder of this handout is the final exam from last year. Keep in mind that it was a 150-minute exam, and this year's exam will be a 24-hour take home.

1. (2 points) *Roller Coaster Problem Revisited*. Recall the roller coaster problem from a homework set:

Suppose there are n passenger processes and one car process. The passengers repeatedly wait to take rides in the car, which can hold C passengers, $C < n$. However, the car can go around the tracks only when it is full.

The following is a solution to this problem:

Shared data: semaphore avail_seat=C, riders=0, seats_taken=0;

Each of the n passenger processes:

```
while (1) {
    wait(avail_seat); // wait in line
    signal(seats_taken); // claim a seat and sit in it
    // now sitting in car waiting for ride to complete
    wait(riders); // wait for car to signal the end of the ride
}
```

The car process:

```
while (1) {
    int counter; // local variable - no mutex needed

    for (counter=0; counter<C; counter++) {
        wait(seats_taken); // wait for each seat to be filled by a passenger
    }
    // go around the tracks
    for (counter=0; counter<C; counter++) {
        signal(riders); // tell this riding passenger that they can leave
        signal(avail_seat); // signal waiting passengers that a seat is open
    }
}
```

```
        // for the next trip around the tracks
    }
}
```

Suppose the semaphore `riders` were to be removed, along with the passenger's call to `wait(riders)` and the car's call to `signal(riders)`. Why does the remaining code fail to provide a solution to the problem? (2 points)

2. (4 points) Deadlock in Distributed Systems. We noted that many systems use the "Ostrich Algorithm" (stick your head in the sand and pretend everything will be just fine) to deal with deadlock. Now consider a distributed system.

- a. Does deadlock become a more important or less important problem? Why? (2 points)
- b. Does it become more expensive or less expensive to detect? Why? (1 point)
- c. Does it become more expensive or less expensive to recover from deadlock? Why? (1 point)

3. (13 points) Distributed Systems.

a. Consider a modern multiprocessing computer system that consists of collection of SMP nodes connected by a fast network. Name two advantages and two disadvantages of presenting the entire system to end users as one large computer as opposed to as a collection of independent computers. (4 points)

b. Suppose you, as a user of this system, wish to run a job that will utilize 4 processors. Name one advantage and one disadvantage of using all four processors on the same node and one advantage of using one processor on each of four nodes. (2 points)

c. Now suppose each node has 25 gigabytes of local disk storage. Would you configure these disks as individual local file systems or would you combine them into a global distributed file system? Why or why not? (2 points)

d. Location transparency is often a desirable feature in a distributed system. To what extent do World Wide Web URLs (*i.e.* `http://www.cs.williams.edu`) provide location transparency? (2 points)

e. In a fully-distributed system, how can we be sure that an event in a process at site A happened before an event in a process at site B? (1 point)

f. How can a multithreaded program run faster than its single-threaded counterpart even on a single-processor system? (2 points)

4. (5 points) Distributed Memory Hierarchy. Recall that the SGI Origin 2000 is a shared memory multiprocessing system. However, memory access is non-uniform, as memory is actually located on boards that contain 2 CPUs each. Those two CPUs can access memory on their local board directly, but for a CPU on one board to access memory on another board, it must be retrieved from the remote memory.

a. Consider such a system with 12 processors (like a small version of the NCSA systems).

The system appears to the user to have a uniform shared memory space. Describe two complications that this adds to the operating system. (2 points)

b. As the user of such a system, under what circumstances would it be helpful to be aware of the underlying memory hierarchy? How might you take advantage of such information? (3 points)

5. (7 points) Network File Systems. Suppose you are a student at a small college in a small town nestled in the scenic Berkshires of western Massachusetts, and you are using your account in the school's Computer Science Lab for your class and project work. Your home directory is located on a central file server and is mounted on all lab workstations, but files from your home directory are *not* cached on local disk.

a. Name a situation where it would be to your advantage to use local disks, which are available as temporary space on each workstation, rather than your home directory, to store files. (2 points)

b. If sufficient local disk space were available on some or all lab workstations, would you recommend that some of it be used as a cache for your home directory? Why or why not? (3 points)

c. Would such a disk cache be safer in a multiuser lab environment than would be a memory cache? Why or why not? (2 points)

6. (4 points) Page Replacement. Consider the following page reference string.

1 3 4 2 5 3 4 1 3 4 6 2 5 1 3

Assume you have 4 frames available, and that they are initially empty. How many page faults will occur if the replacement algorithm is...

a. First In First Out (1 point)

b. Least Recently Used (1 point)

c. Optimal (1 point)

d. Can an algorithm that exhibits Belady's Anomaly be optimal? (1 point)

7. (9 points) File Structures. Consider a file allocation scheme using a file allocation table similar to the DOS/Windows FAT16 scheme. Such a filesystem contains a fixed maximum size of the file allocation table, which determines the number of addressable blocks/clusters.

a. If the file allocation table contains 65536 (64K) entries, and the block size is 8Kbytes, what is the maximum size of a filesystem? (2 points)

b. If the block size is increased to 32Kbytes, what is the maximum size of a filesystem? (1 point)

c. Suppose you have a disk equal in size to your answer in part (b). Name one advantage and one disadvantage to partitioning the disk into four partitions with a cluster size of 8K vs. one partition with a cluster size of 32K. (2 points)

- d. How does a filesystem using UNIX inodes provide efficient direct (random) access within files? (1 point)
 - e. It has been suggested that the first part of each UNIX file be kept in the same disk block as its inode. What good would this do? (2 points)
 - f. Name one reason why a file extension may be insufficient to identify the type of a file. (1 point)
8. (6 points) Security
- a. It has been argued that an open source system such as Linux or FreeBSD can both help and hinder security efforts when compared to a system where source code is less likely to be available on a widespread basis. What do you think? Justify your answer briefly. (3 points)
 - b. Name a feature that could be added to a C compiler could eliminate a large number of security holes. Why is it not implemented? (2 points)
 - c. Suppose you are a UNIX system administrator and some of your users have discovered the algorithm used to encrypt passwords on the system. What should you do? (1 point)