

Problem Set 2

Due Mar 4th at 6pm (bonus if submitted before Fri Mar 2nd at 6pm)

Name: _____

Recitation (# or time): _____

Hours (not including your first reading of the chapters)
required to complete problem set: _____

Please type your answers and submit a PDF where the filename has the following format:
PS2-<lastname>-<firstname>.pdf

1. (10 pt) Describe and explain each of the following (use examples):
 - a. the concept of interleaving instructions of two threads
 - b. Each of the requirements for deadlock (give examples). Explain why the first three are necessary but not sufficient for deadlock.
2. (10 pt) Compare and contrast the following (make sure to define each of the items):
 - a. mode switch vs process switch
 - b. semaphore vs conditional variable
3. (20 pt) Given the code below answer the following questions (explain your reasoning):
 - a. Is the function thread safe?
 - b. Is the function reentrant?

```
int temp;
void swap(int *y, int *z)
{
    int local;

    local = temp;
    temp = *y;
    *y = *z;
    *z = temp;
    temp = local;
}
```

Problem Set 2

Due Mar 4th at 6pm (bonus if submitted before Fri Mar 2nd at 6pm)

4. (30 pt) Barrier Synchronization: A **barrier** is a tool for synchronizing the activity of a number of threads. When a thread reaches a **barrier point**, it cannot proceed until other threads have reached this point as well. When the last thread reaches the barrier point, all threads are released and can resume concurrent execution.

You have a number of threads processing data and each piece of data takes a different amount of time to process. When a thread has completed its processing, it will ask for more work. However, the data requires that 4 workers all start together on the next 4 pieces of data.. This means we want to gather 4 threads before having them begin processing.

Please provide a solution (pseudocode), explain how it works, show that it is starvation free and deadlock free.

5. (30 pt) CU wants to show off how politically correct it is by applying the U.S. Supreme Court's "Separate but equal is inherently unequal" doctrine to gender, ending its long-standing practice of gender-segregated bathrooms on campus. However, as a concession to tradition, it decrees that when a woman is in the bathroom, other women may enter, but no men, and vice versa. Also, due to fire code, at most N ($N > 1$) individuals may use the bathroom at any time. Your task is to write two functions: `man_use_bathroom()` and `woman_use_bathroom()`. Provide a monitor-based solution that manages access to the bathroom. Your solution should be fair, starvation free and deadlock free.