# CSSE 332 -- Operating Systems

## Introduction to Mutual Exclusion

Name: _____

**Question 1**. (5 points) In your own words, describe the main problem leading the sample code in `simple_example.c` to produce incorrect results.

**Question 2**. (5 points) In the space below, write down the main API functions used to create and use mutex lock in the `pthreads` library.

**Question 3**. (10 points) The code listing below contains a major bug, identify it and suggest a way to fix it.

```c
void *thread1(void *ignored) {
  // some code
  pthread_mutex_lock(&lock);
  // do some stuff

  pthread_mutex_unlock(&lock);
  return 0;
}
```

```c
void *thread2(void *ignored) {
  // some initialization code
  pthread_mutex_unlock(&lock);
  pthread_mutex_lock(&lock);
  // do some stuff
  pthread_mutex_unlock(&lock);
  return 0;
}
```

**Question 4**. Consider the following code snippet:

```c
struct metadata {
  unsigned int id, length;
  int *array;
};

void *thread_function(void *arg) {
  struct metadata *ptr = (struct metadata*)arg;
  printf("Thread %d started, processing array of length %d...", ptr->id,
         ptr->length);

  // swap_max_with_last is a function defined elsewhere that finds the
  // maximum element in an array and swaps it with the last element in that
  // array (the last element is specified by the argument end below).
  swap_max_with_last(ptr->array, 0, ptr->length - ptr->id + 1)

  printf("Thread %d done.\n", ptr->id);
  return 0;
}

int main(int argc, char **argv) {
  int *array = malloc((2<<20) * sizeof(int));
  struct metadata all_meta[TOTAL_THREADS];
  pthread_t threads[TOTAL_THREADS];
  int i;

  // defined elsewhere
  initialize_array(array);
  for(i = 0; i < TOTAL_THREADS; i++) {
    all_meta[i].id = i + 1;
    all_meta[i].length = 2<<20;
    all_meta[i].array = array;
    pthread_create(&threads[i], NULL, thread_function, &all_meta[i]);
  }

  for(i = 0; i < TOTAL_THREADS; i++) {
    pthread_join(threads[i], NULL);
  }
  exit(0);
}
```

(a) (5 points) What do you think this piece of code is attempting to do?

(b) (10 points) In the code listing above, identify any critical sections and possible race conditions. Feel free to add your notes to the code listing itself.

(c) (5 points) At the end of the `main`, what do you expect the contents of `array` to be?