HW 2: Review of Chapter 5
Due: 3:20pm, Wednesday Oct 12, 2011

Problem 1 (30 points) Jurassic Park consists of a dinosaur museum and a park for safari riding. There are \( m \) passengers and \( n \) single-passenger cars. Passengers wander around the museum for a while, then line up to take a ride in a safari car. When a car is available, it loads a passenger and rides around the park for a random amount of time. If the \( n \) cars are all out riding passengers around, then a passenger who wants to ride waits; if a car is ready to load but there are no waiting passengers, then the car waits. Solve this problem using semaphores.

Problem 2 (40 points) This problem demonstrates the use of semaphores to coordinate three types of processes. Santa Claus sleeps in his shop at the North Pole and can only be awakened by either (1) all nine reindeer being back from their vacation in the South Pacific, or (2) some of the elves having difficulties making toys; to allow Santa to get some sleep. The elves can only wake him when three of them have problems. When three elves are having their problems solved, any other elves wishing to visit Santa must wait for those elves to return. If Santa wakes up to find three elves waiting at his shop’s door, along with the last reindeer having come back from the tropics, Santa has decided that elves can wait until after Christmas, because it is more important to get his sleigh ready. (It is assumed that the reindeer do not want to leave the tropics, and therefore they stay there until the last possible moment.) The last reindeer to arrive must get Santa while the others wait in a warming hut before being harnessed to the sleigh. Solve this problem using semaphores.

Problem 3 (10 points) What is the difference between \texttt{csignal} and \texttt{cnotify} primitives in a monitor?

Problem 4 (20 points) Another atomic machine instruction that supports mutual exclusion that is often mentioned in the literature as the \texttt{test\&set} instruction, defined as follows:

```c
boolean test_and_set (int i)
{
    if (i == 0) {
        i = 1;
        return true;
    } else return false;
}
```

Define a procedure similar to those of Figure 5.2 that uses the \texttt{test\&set} instruction.