Operating Systems Course	**Deadline**: 28 April 2012
Assignment 6: IPC	**Instructor**: Hossein Momeni
Golestan University	

Problem1

Three processes are running on a computer system, namely P1, P2 and P3. Due to the following table they run Up and Down operators on a semaphore S. If two processes are blocked and an Up operator is called, then the blocked process with the greater index would be run. What's the final state of these tree processes if they run the operators according to this table? (from left to right)

Process	P1	P2	P3	P2	P1	P3	P2	P2	P3	P1
Command	Down	Down	Up	Down	Up	Up	Down	Down	Up	Down

Problem2

Consider the swap as an atomic procedure Lock is a global variable with initial value equal to false. Analyze the following solution for critical section.

```
repeat
        key := true;
        repeat
                swap(Lock,key);
        until key=false;
        C.S.
        Lock:=false
        R.S.
Until false
```

Problem3

Solve Producer-Consumer problem **using message passing** when the buffer is infinite.

Problem4

Is the following a correct solution to the critical section problem? Argue whether or not the alleged solution ensures **mutual exclusion**, **progress**, and **bounded waiting**.

```
boolean blocked[2] = {false, false};
int turn = 0;

thread t0()
{
  while(true)
  {
    blocked[0] = true;
    while (turn != 0)
    {
      while (blocked[1]) { }
      turn = 0;
    }
    CRITICAL SECTION;
    blocked[0] = false;
    REMAINDER SECTION;
  }
}

thread t1()
{
  while (true)
  {
    blocked[1] = true;
    while (turn != 1)
    {
      while (blocked[0]) { }
      turn = 1;
    }
    CRITICAL SECTION;
    blocked[1] = false;
    REMAINDER SECTION;
  }
}
```

Problem 5.

Suppose we have a producer and a consumer. The producer produces items and inserts them into a queue owned by the consumer, while the consumer consumes items from its queue in FIFO order.

Each item requires 1 time quantum to produce and 1 time quantum to consume. The queue is initially empty and has a maximum size of 3, If the queue is full when a producer wants to run, the producer will spin wait until the queue is not full. If the queue is empty when the consumer wants to run, the consumer will spin wait until the queue is not empty.

Consider the case when we run three producers P1, P2, and P3, and one consumer C1 and all of the processes are runnable starting at time zero.

a) If Round-Robin scheduling is used to execute the processes, how many items will each process have produced and consumed at the end of 10 time quanta? Assume that the initial run queue order is P1, P2, P3, C1.

b) If priority scheduling is used to execute the processes and the priorities P(X) are assigned as P(C1)=4, P(P3)=3, P(P2)=2, P(P1)=1, how many items will each process have produced and consumed at the end of 10 time quanta? Assume that a larger number is a higher priority.

Problem6.

Solve the Philosopher problem with monitor mechanism.