Homework 3

Assigned on 10/20/05 due 10/27/05 11:59:59 PM. Please submit a typewritten solution and email it to your TA by the deadline.

Total points = 50 points

**Question 1.** (10 points) A sender process $P$ sends a sequence of messages to a receiver process $Q$. Each message $m$ is stamped with a sequence number $\text{seq}$. If channels can lose or reorder messages, then is it possible to design a protocol for FIFO message delivery using bounded $\text{seq}$? Do not worry if the receiver accepts duplicate copies of the same message.

**Question 2.** (10 points) In a spanning tree of a graph, there is exactly one path between any pair of nodes. If a spanning tree is used for broadcasting a message, and a process crashes, some nodes will not be able to receive the broadcast. Our goal is to improve the connectivity of the subgraph used for broadcast, so that it can tolerate the crash of one process.

What kind of minimal subgraph would you use for broadcasting, so that messages will reach every process even if one process fails? Suggest a distributed algorithm for constructing such a subgraph. How many messages will you need to complete a broadcast?

**Question 3.** (10 points) A sender process $P$ sends a sequence of messages to a receiver process $Q$. Each message $m$ is stamped with a sequence number $\text{seq}$ that increases monotonically. The program for $P$ can be specified as follows:

```plaintext
define seq : integer {initially seq = 0}
do true ⇥ send m[seq] to Q; seq := seq + 1 od
```

In the absence of failures, $Q$ receives the messages in the same order in which they are sent. Failures may cause messages to reach $Q$ out of order, but messages are never lost,
and it is important that Q accepts the message is the same order in which they were transmitted.

(a) Describe the program for Q. Calculate its buffer requirement.

(b) Now assume that Q has a buffer that can hold at most one message. Rewrite the programs of P and Q, so that Q accepts the messages in the same order in which P sent them. Argue why your solution will work.

**Question 4.** (10 points) The topology of a distributed system is represented by an undirected graph $G=(V, E)$, Assume that each process is fail-stop, and each fail-stop failure can be detected by its neighbors. If a spanning tree algorithm running on $G$ computes a spanning tree $(V’, E’)$, and a fail-stop failure is detected, then propose how the neighbors of the faulty process will initiate a reconfiguration of the spanning tree. You can assume that $G$ is k-connected ($k >1$). Argue why your solution will work.

**Question 5.** (10 points) Consider Peterson’s algorithm for leader election on a unidirectional ring of sixteen processes 0 through 15. Show two sample cases: in the first, the leader will be elected after one round (this is the best case), and in the second, leader election will take the maximum possible number of rounds. What is this number?