T-106.420 Concurrent programming
Homework assignment 3 (Oct. 27, 2003)

The deadline for this exercise is Sunday, Nov. 9, 2003

1. The following is an old exam question:
   (The Bear and the Honeybees) Given are N honeybees and a hungry bear. They share a pot of honey. The pot is initially empty; its capacity is H portions of honey. The bear sleeps until the pot is full, then eats all the honey and goes back to sleep. Each bee repeatedly gathers one portion of honey and puts it in the pot; the bee who fills the pot awakens the bear. Represent the bear and honeybees as processes and develop a code that simulates their actions. Use semaphores for synchronization. Show the declarations and initial values of all semaphores and variables that you use.

   Part (a) and (b) present two code segments that were submitted as a solution in 2002. They are both wrong. You are asked to explain why. In part (c) you are asked to present a correct solution.

   (a) *What is wrong with the following solution?*

```
int h = 0;
sem mutex = 1;
sem full = 0;

process Honeybee[i = 1 to N] {
  while (true) {
    P(mutex);
    h++;
    if (h == H)
      V(full);
    V(mutex);
  }
}

process Bear {
  while (true) {
    P(mutex);
    P(full);
    h = 0;
    V(mutex);
  }
}
```

(b) *What is wrong with the following solution?*

```
int h = 0;
sem full = 0;
sem pot = 0;

process Bear {
  while (true) {
    P(full);
    h = 0;
    V(full);
    V(pot);
  }
}
process Honeybee[i = 1 to N] {
  while (true) {
    P(pot);
    h++;
    if (h == H)
      V(full);
    else
      V(pot);
  }
}
```

(c) *Present a correct solution to the problem.*

2. Suppose you are writing programs for a machine that has atomic increment and decrement instructions. INC(x) atomically adds 1 to integer variable x, and DEC(x) atomically subtracts 1 from integer variable x. Develop a busy-waiting implementation of P(s) and V(s). *You may assume that INC(x) and DEC(x) return the modified value of x.*

3. Use a monitor to implement an ordinary semaphore. You answer must show the declaration and initialization of all variables and condition variables you use within the monitor, as well as all monitor procedures that are necessary to implement the semaphore operations.

Important note related to the 4th home assignment:
The 4th home assignment is related to the Message Passing Interface (MPI), and it is a programming assignment. A collection of MPI related links and requirements will be presented on the course web–page soon, and will be announced in the course news–group. Get familiar with MPI in advance in order to avoid having to spend too much time with learning the basics during the assignment. The allowed time to solve the 4th assignment will be longer, students are allowed to submit their solution until Wednesday, 7th of January, 2004.