Homework #1

Due Date: January 16, 2000 **Points**: 60

1. (*20 points*) In the example of virtual machines, with a compiler above an operating system above two levels of virtualizing kernel, how many privileged instructions would be executed at each level if the instruction executed by the compiler can be emulated without use of privileged instructions by the operating system?
2. (*20 points*) Is the following program properly nested? Please either show that it is by rewriting the program using **parbegin** … **parend**, or prove that it is not properly nested. (The S*i* are statements.)

```
        c4 := 2;
        c6 := 2;
        S1;
        fork p1;
        S3;
        fork p2;
        S5;
        goto p4;
p1:     S2;
        goto p2;
p2:     join c4, p3;
        quit
p3:     S4;
p4:     join c6, p5;
        quit
p5:     S6
        quit
```

3. (*20 points*) Synchronization within monitors uses condition variables and two special operators, **wait** and **signal**. A more general form of synchronization would be to have a single primitive, **waituntil**, that had an arbitrary Boolean predicate as parameter. Thus, one could say, for example,

   waituntil $x < 0$ **or** $y + z < n$

   The **signal** primitive would no longer be needed.

   a. Use this more general form to solve the producer-consumer problem.
   b. Is this construct more, less, or as, powerful as using wait and signal (in Hoare's version of monitors)?
   c. Why do you think it is not used in practice?