CMPSCI 377: Operating Systems

Homework 3: Monitors and Deadlock

Due: October 26, 1999

VIP Students: Due 12 days from when you receive the assignment.

---

1. (20 pts) **Semaphores & Monitors.**
   (a) (16 pts) Solve the candy shop problem with semaphores and with monitors. In the candy shop, a customer enters and takes a number Some number of sales people wait on them in order of their arrival. If a sales person is free when the customer takes a number, the sales person waits on the customer right away (assume a FIFO queue on the wait queue). If no sales person is free when the customer takes a number, the customer waits. Write `Enter`, and `Finish` routines for the customers.
   (b) (4 pts) Which of your solution is easier to understand and thus get correct?
2. (10 pts) **Monitors.** Write a monitor solution to the north-south tunnel problem. Suppose a two-way (north-south), two-lane road contains a long one-lane tunnel. A southbound (or northbound) car can only use the tunnel if there are no oncoming cars in the tunnel. Because of accidents, a signaling system has been installed at the entrances to the tunnel. When a car approaches the tunnel, a sensor notifies the controller computer by calling a function `arrive` with the car's travel direction (north or south). When a car exits the tunnel, the sensor notifies the controller computer by calling `depart` with the car's travel direction. The traffic controller sets the signal lights: green means go, and red means stop. Construct an algorithm for controlling the lights such that they operate correctly even when most cars approach the tunnel from one direction. In the monitor solution, consider using a `CVar.broadcast` to release all the waiting cars from the north (or south) at once. In which cases, does this make sense?
3. (10 pts) **Deadlock.** Short answer questions:
   (a) A system has six tape drives (a, b, c, d, e, f), with n processes competing for them. Each process may need two of the drives. For what values of n is the system deadlock free?
   (b) Can a system be in a state that is neither deadlocked nor safe? If yes, give an example system.

4. (10 pts) **Deadlock.** Consider the following system snapshot using the data structures in the Banker's algorithm, with resources A, B, C, and D, and processes P_0 to P_4.

	Allocation				Max				Available				Need			
	A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
									3	2	1	0				
P_0	3	0	0	2	6	0	1	2								
P_1	1	0	0	0	1	7	5	0								
P_2	1	3	5	4	2	3	5	6								
P_3	0	6	3	2	1	6	5	2								
P_4	0	0	1	4	1	6	5	6								

Using Banker's algorithm answer the following questions.

(a) How many resources of type A, B, C, and D are there?

(b) What is the content of the *Need* matrix?

(c) Is the system in a safe state? Why?

(d) If a request from process P_4 arrives for additional resources of (1,2,0,0), can the Banker's algorithm grant the request immediately? Show the new system state, and other criteria.