Solutions
15-410, Fall 2008, Homework Assignment 1.

1 Nuts & Bolts (10 pts.)

What went wrong?

It seems as if somebody stuck a bad value into `%DS`!

Note that you can run for "quite a while" with a bad value in a segment register. In this case, execution was able to continue through the stack operations involved in returning from `foo()`, invoking `SIM_printf()` (aka `lprintf()`, invoking `vsnprintf()`, and invoking `_doprnt()`. What failed was the first "data" reference, namely checking the first byte of the string against the null character.

2 Dining Philosophers (10 pts.)

Explain why, at any table with a mixture of left-handed and right-handed philosophers, deadlock cannot occur.

First observe that due to the pattern of resource acquisition around the table there is only one possible wait-cycle, i.e., the cycle involving all philosophers.

So assume such a deadlock. Every philosopher must be waiting, by which we mean waiting for a chopstock held by another. Thus all chopsticks must be held.

If our table has a mixture of left-handed and right-handed philosophers, there must be at least one pair of adjacent philosophers of opposite handedness.

First assume a left-handed philosopher ("Lenny") seated to the left of a right-handed philosopher ("Roger"). Because all chopsticks must be held, either Lenny or Roger must hold the chopstick which is between them. If Lenny holds it, that is his "right chopstick," so he must previously have acquired the chopstick to his left. Thus Lenny holds two chopsticks, and is not waiting to acquire any (he is eating), so there is not a deadlock. If Roger holds the chopstick which is between them, that is his "left chopstick," so he must previously have acquired the chopstock to his right, etc.

If we call the previous case the "outward reaching" case, then the "inward reaching" case is a right-handed philosopher seated to the left of a left-handed philosopher. Again, if we have a deadlock, all chopsticks must be held, so one of them holds the chopstick between them. This means that the other one holds no chopsticks, since the one between them is the first one each will try to acquire (either Roger reached right and got it, or Lenny reached left and got it; the other is waiting for it before reaching in the outward direction). If there are n philosphers and n chopsticks, all chopsticks held, but one philosopher holds zero chopsticks, then $n-1$ philosophers hold n chopsticks. Some philospher must hold two chopsticks; that philosopher is not waiting (he is eating), so there is no deadlock.

Don't worry if your proof is much shorter than this one—brevity is the soul of wit.

3 Zero Terror in the Bakery (20 pts.)

In this trace, each box is an action ("enter," "leave," or "=") or the observation of a fact ("!c[0]" will means we'll skip out of a "while(c[0])" loop).

Execution Trace

time	Thread 0	Thread 1	Thread 2
0		...	
1	c[0] = T	...	
2	n[0] = ffff	leave	
3	c[0] = F	n[1] = 0	
4		c[1] = T	c[2] = T
5		n[1] = 0	n[2] = 0
6		c[1] = F	c[2] = F
7	!c[0]		
8	(ffff,0) !> (ffff,0)		
9	!c[1]		
10	!(n[1]!=0)		
11	!c[2]		
12	!(n[2]!=0)	!c[0]	!c[0]
13	enter	n[0]!=0	n[0]!=0
14	leave	n[0]!=0	n[0]!=0
15	n[0] = 0	n[0]!=0	n[0]!=0
16		!(n[0]!=0)	!(n[0]!=0)
17		!c[1]	!c[1]
18		!(n[1]!=0)	!(n[1]!=0)
19		!c[2]	!c[2]
20		!(n[2]!=0)	!(n[2]!=0)
21		enter	enter
22		uh-oh	uh-oh