Chapter 9

Semaphore Practice Program:

Build a solution to the dining philosophers problem as described in lecture. You should have a main process that sets up the semaphores; then forks 5 philosopher processeses (children); then waits for the 5 philosopher processes to finish; then removes the semaphore.

Your philosophers should do the following steps 5 times (loop):
print "philosopher xxxx eating" (Use the process PID in place of xxxx)
eat (sleep) for 1 second
print "philosopher xxxx thinking"
think (sleep) for 4 seconds

A starter file as available as `/net/326/philosphers.c`

This problem is ungraded but is good practice for using semaphores.

Homework Program:

Build a multiplier using pipes.

You may use C++ for this if you wish (but are NOT required to!). Be sure to rename the file `pipe.cpp` if you do. The compiler is called `g++` and works the same.

You will have a parent process that does all the input and output and a child process that does the multiplies.

You will need two pipes, one to send from the parent to the child the other to send from the child to the parent.

The parent process does the following 5 times (loop):
cin the multiplicand
write the multiplicand to the pipe.
cin the mulitplier
write the multiplier to the pipe.
read back the answer from the other pipe

Child process behavior
read the multiplicand from the pipe
read the multiplier from the pipe
multiply
write the answer to the other pipe

if the read returns 0 (EOF) the child should exit.

The last example from lecture is available as `/net/326/pipe.c`

Demo: Your pipe program and show the instructor your code.