Program 1:

Build a multiplier using named pipes (aka FIFOs).

You will have a client process that does all the input and output and a server process that does the multiplies. C++ is acceptable here.

You will need `two` named pipes, one to send from the client to the server the other to server from the client.

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

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

The server process exits on control-c

The server should be started first.

`fifo.c` and `fifor.c` are available from `/net/326`.

When you are done. Remove the fifo (by hand using the `rm` command).

Program 2:

Build a multiplier using messages.

You will have a client process that does all the input and output and a server process that does the multiplies. C++ is acceptable here.

You will need two message queues, one to send from the client to the server the other to server from the client.

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

Server process behavior
read the multiplicand from the message queue
read the multiplier from the message queue
multiply
write the answer to the other message queue

The server process exits on control-c

`sndmes.c` and `rcvmes.c` are available.

Demo: Your fifo and message programs.