

New York University

Tandon School of Engineering

Department of Computer Science and Engineering

Introduction to Operating Systems
Fall 2024

Assignment 6
(10 points)

- 1) (8 points) Repeat assignment 5B, except that you shall now use a TCP/IP socket for communicating between the processes instead of a pipe.

Use the following socket functions in their default mode. You may use the `man` command in your Linux virtual machine for information about the parameters:

CLIENT	SERVER
<code>socket()</code> – opens a socket (similar to <code>pipe()</code>)	<code>socket()</code>
<code>connect()</code> – connects to a server	<code>bind()</code> – assigns a particular port number to the server <code>listen()</code> – listens to connection requests from clients <code>accept()</code> – accepts a connection from client
<code>read()</code> – reads a buffer from the socket, just as in file or pipe reading	<code>write()</code> – writes a buffer to the socket, just as in file or pipe writing
<code>close()</code> – closes the socket	<code>close()</code>

You shall use sockets of type `SOCK_STREAM` and assign the parent (consumer) as the client and the child (producer) as the server.

Insert an initial random wait (1 to 3 seconds) at the child process (but not the parent) prior to it starting to listen and accept connections.

The parent process (client) may thus fail to connect if it tries to do so before the child process (server) has started to listen (which is after the random wait). As such, you should insert a loop in the parent that repeatedly attempts to connect, waiting 100 ms between attempts, till it succeeds, eventually.

- 2) 2 points): Answer the following **for part 2**:
- Which of the calls above are blocking and which are not? Explain what that means?
 - Is this a form of direct communications or indirect communications?
 - What is the failure flag returned from `connect()` that indicates the server is not ready?
 - How would you change your program to communicate between processes in different machines?

What to submit:

Please submit the following files individually:

- Source file(s) with appropriate comments.
The naming should be similar to “**lab#_\$.c**” (**#** is replaced with the assignment number and **\$** with the question number within the assignment, e.g. lab4_b.c, for lab 4, question c OR lab5_1a for lab 5, question 1a).

- 2) A single pdf file (for images + report/answers to short-answer questions), named “**lab#.pdf**” (# is replaced by the assignment number), containing:
 - Screen shot(s) of your terminal window showing the current directory, the command used to compile your program, the command used to run your program and the output of your program.
- 3) Your Makefile, if any. This is applicable only to kernel modules.

RULES:

- You shall **use kernel version 4.x.x or above**. You shall not use kernel version 3.x.x.
- You may consult with other students about GENERAL concepts or methods but copying code (or code fragments) or algorithms is NOT ALLOWED and is considered cheating (whether copied from other students, the internet or any other source).
- If you are having trouble, please ask your teaching assistant for help.
- You must submit your assignment prior to the deadline.