In this exercise we familiarize ourselves to interprocess communication (IPC).

Exercise 10 (Server with unix domain sockets, 1p)

Write a server that uses Unix domain sockets. The address of the server is server.sock. Clients send a service request, which is an ascii string containing one character, one space and integer number indicating a time interval in seconds and a terminating zero (total four characters). The request means that the server needs to send the character ten times. Time interval between sendings is the time in the request.

The requirements for the server are:

1. The server needs to serve all clients "simultaneously"
2. When the server has sent a character 10 times to the client and when the client has read them, the next read must cause end of file.
3. The server is running continuously and you can start the testing program lab_client.exe (described below) again and again.
4. No zombies are born when the server is running.

It is allowed to use whatever technique in the implementation.

Use the program lab_client.exe to test your server (available from Tuubi). This program generates four clients. The request of the first client is a 1 (send character a 10 times in one second intervals). The request of the second client is b 2 (send character b 10 times in two seconds intervals) etc. Clients only write the results immediately to the screen without any spaces or newlines. When end of file has been received, then client writes (eof). Use for example two terminal windows and run your server in one and the test program in another.

Your server works correctly, if the output looks something like

cadbaabcadbaacbaadbacab(eof)cdbbcdbcbd(eof)ccdcd(eof)dd(eof)

and a is displayed once per second, b is displayed twice per second and so on. (eof) is displayed in 10 seconds intervals.

Remark. Unix domain sockets are used. The principle of the server is exactly the same even if the server were used over the network. The only difference is in the addresses, in the definitions of the address structures and in the initialization of structure members. All phases that are needed to create connections and to pass information are similar.