

Exercise Sheet 5

In this exercise sheet you will practice interprocess communication with pipes.

Task 1

Imagine a complex system consisting of multiple servers: database server, Middle-ware server, and web server. These servers log different information by using named pipes (FIFO). In order to print the output generated by any of these servers you decide to create your own service. This service will read from any of the aforementioned FIFO (read only when something is available) and will print the information on the standard output.

For this task your goal is to simulate the database server, middle-ware server, and web sever as well as to implement the requested logging service.

First of all, you need to create 3 named pipes, each of them representing one of the aforementioned servers.

To simulate each of the programs, you need to create a program which runs forever in an endless loop. Within this loop, a quantity between 2 and 7 seconds is randomly generated. The program should sleep then for that quantity of time. After that it should log a message (you choose the message content! Keep your messages shorter than `PIPE_BUF` bytes to ensure atomic writes of your messages) on its corresponding fifo.

The login service will open the three fifos and will wait for input on all of them using `select()`. Each received message in any of these pipes should be consumed by the service and printed on the standard output. Assume that this service will also run forever in an endless loop.

An example of how the output will be shown could be:

```
[web] message 1 from web server
[middle-ware] message 1 from middle-ware server
[middle-ware] message 2 from middle-ware server
[web] message 2 from web server
[database] message 1 from database
[middle-ware] message 3 from middle-ware server
```

Task 2

Create a program that implements the execution of the command `ls | grep <keyword>`. To implement this behaviour, your program is required to create a proces that executes the command `ls`. Create another process that executes

the command `grep keyword`. Communicate between both process by using an unnamed pipe. This assignment **requires** that you explicitly use `fork()`, `pipe()`, and `dup()/dup2()`.