CS307 - Operating Systems — 2021-2022 Fall

# Programming Assignment 1 Report

Doğukan Yıldırım — 28364

## 1 Command and Options

```
man ping | grep -A 2 -e '-w' > output.txt
```

'ping' command sends one or more ICMP ECHO_REQUEST packages to specified network hosts to check whether the specified host or network is available. It is mainly used for troubleshooting and network connectivity tests.

'-w' (deadline) option is used to specify the number of seconds before the ping exits, independent of how many packets have been transmitted or received.

I picked 'ping' command because Computer Networks has always been an interest of mine (I am hopefully going to take that course next semester); how did the first computers communicate in the first place, how the internet came to be, and how can we transmit large amounts of data through vast distances in seconds... these are fascinating topics.

I picked '-w' option because in the man page, I saw the most dreadful word right next to it: **deadline**. Jokes aside, specifying how many seconds before the ping exits seemed like an important concept to me. For example, if I wrote a program that sends get requests, I would not want to be stuck, waiting forever for a response, so it could be a good practice to specify a timeout variable.

I have picked 'grep' options '-A 3' to print 3 lines after the match and '-e' to solve the problem of including the special character '-'.

## 2 Process Hierarchy

**The process hierarchy is as follows:**

The main function will be the shell process. Firstly, the shell process will utilize pipe system call so that processes can communicate. We will refer to the pipe as 'fd'. During the shell process, **fork()** will be called inside the shell process to create a new process with the process id of **cpid_1**.

Inside this child process, **cpid_1**, fd[1] file descriptor will be duplicated to STDOUT_FILENO so that the output of the **man** command will be written to the pipe. Both read and write ends will be closed. After that, **man** command with variable 'ping' will be executed. The shell process will wait for the child process, **cpid_1**, to finish.

After **cpid_1** finishes, **fork()** will be called again inside the shell process to create a new process with the process id of **cpid_2**.

Inside this child process, **cpid_2**, fd[0] file descriptor will be duplicated to STDIN_FILENO so that the input of the **grep** command can be read from the pipe. Both read and write ends will be closed. Before the execution of the **grep** command, a new file 'output.txt' will be opened, and the file descriptor of this file will be duplicated to STDOUT_FILENO so that the output of the **grep** command can be written to the 'output.txt' file. Afterwards, 'grep' command with variables '-A 2', '-e', '-w' will be executed. The shell process will wait for the child process, **cpid_2**, to finish.

After **cpid_2** finishes, **the execution will be completed**, and the shell process will terminate.