**آرمان حیدری – 97521252 - سوال 1**

**الف)**

هدف در این سوال این است که دو command را پشت سر هم انجام دهیم و در واقع خروجی دستور اول را به جای آن که چاپ کنیم به دستور دوم بدهیم. برای این کار دو process میسازیم که فرزند مسئول اجرای دستور اول است و پدر دستور دوم.

پس در قسمتی که process فرزند اجرا میکند، (داخل شرط صفر بودن process id ) باید دستور اول را execute کنیم. سپس برای آن که خروجی آن استاندارد شود و به جای چاپ شدن، به یک سر pipe، map شود از dup استفاده میکنیم. Fd[1] را برای فرزند که درواقع write میکند و fd[0] را برای پدر که در واقع read میکند در نظر میگیریم. به همین صورت در process پدر هم با استفاده از exec فیلتر روی خروجی دستور اول را انجام میدهیم. باید حواسمان به بستن file descriptor ها در محل مناسب هم باشد.

**ب)**

File Descriptions are a user-defined string that can be assigned to a file or folder, often used to keep notes about the file contents.

In [Unix](https://en.wikipedia.org/wiki/Unix) and [Unix-like](https://en.wikipedia.org/wiki/Unix-like) computer operating systems, a file descriptor(FD) is a unique identifier ([handle](https://en.wikipedia.org/wiki/Handle_(computing))) for a [file](https://en.wikipedia.org/wiki/File_(computing)) or other [input/output](https://en.wikipedia.org/wiki/Input/output) [resource](https://en.wikipedia.org/wiki/System_resource), such as a [pipe](https://en.wikipedia.org/wiki/Pipe_(Unix)) or [network socket](https://en.wikipedia.org/wiki/Network_socket).

The pipe() function shall create a pipe and place two file descriptors, one each into the arguments fildes[0] and fildes[1], that refer to the open file descriptions for the read and write ends of the pipe. Their integer values shall be the two lowest available at the time of the pipe() call.