Unix Commands – Exercises

1. Write a shell script which will contain the following (independent) commands:

* a command which lists all writable files
* a command which lists all shell scripts (.sh files)
* a command which lists all files with names starting with task, followed by exactly one number and optional letters, and ending with .sh (e.g. task1.sh, task2a.sh, task3.sh)
* a command which lists all directories with names containing exactly 8 characters, starting with 2 capital letters, followed by small letters or digits and ending with a digit (e.g. OSt1gr01, OSkol2t1, OSk2gr03)
* a command which lists all files and directories with names of length 10 (e.g. Operativni, zad1t1a.sh)

ls -l | grep ^..w

ls -l | grep .\*\.sh$

ls | grep ^task[0-9][a-zA-Z]\*\.sh$

ls -l | grep ^d | awk '{print $10;}' | grep '[A-Z]\{2\}[a-zA-Z0-9]\{5\}[0-9]$'

ls | grep '[0-9a-zA-Z]\{10\}'

1. You have a register of servers, given in the file servers.txt. Write a single UNIX command which will print the IP address of the server submitted as a first command line argument. The names of the servers are unique and don’t contain whitespaces.

Content of "servers.txt":

OS-server 194.149.136.94

FINKI-website 194.149.137.199

Courses 194.149.137.160

For the command line argument "Courses", the shell script prints "194.149.137.160".

cat servers.txt | grep "Courses" | awk '{print $2;}'

1. Write a single UNIX command which will print out the number of unique users which have logged in on the OS server in the last several days. You only need to consider the logins from IP addresses which don’t belong to FINKI’s laboratories. The addresses of FINKI’s laboratories are in the following format: 10.10.X.Y.

Change to any IP you like

last | grep '[0-9]\{6\} \*pts/[0-9]\{2\} \*92' | awk '{print $1;}' | uniq | wc -l

1. Using pipes, redirection and / or text filtering, write a single UNIX command which will count how many students are currently using the command man. For the counting, use the list of all active processes for all users of the OS server. The result of the command should be a number. Students are those users of the server which have a username of exactly 6 digits.

ps -ef | grep man | awk '{print $1;}' | wc -l

1. Using pipes, redirection and / or text filtering, write a **single** UNIX command which will count how many students are currently working in nano (have nano as an active process). For the purpose of counting, use the list of all active processes for all users of the OS server. The result of the command should be a number. Students are the users of the server which have a username of exactly 6 digits.

ps -ef | grep nano | awk '{print $1;}' | grep '[0-9]\{6\}'

1. Using pipes, redirection and / or text filtering commands, write **a single UNIX command** which will find the longest session of a user on the server. The session has to be ended. The command should print the session length and the username of the user (e.g. 05:36 111211). If multiple sessions fulfill the criteria, it's sufficient to print only one.

last | awk '{print $1, $10;}' | grep '[0-9]\* (' | sed -e 's/(//g' -e 's/)//g' | sort -k 2 -r | head -1