Tasks should be executed on educational server

1 Create a text file named “hello-world” which contains 3 lines:
Hello
Wonderful
World
Specify how you created it.

echo -e "Hello\nWonderful\nWorld" > ~/hello-world

This command uses echo to print the desired content with newline characters (\n) between the lines, and the output is redirected to a file named "hello-world" using the > operator.

The -e option for echo enables the interpretation of escape sequences, allowing the use of \n for newline.

2. Create a new file with the following content, based on the content of “hello-world”:
World
Wonderful
Hello
It should be created without using echo, text editors, cat > new_file and so on, including script languages like AWK or Python. To do this, remind all recently learned text processing tools learned. The file can be created by using multiple commands.

tail -1 ~/hello-world > ~/new_file && head -2 ~/hello-world | tail -1 >> ~/new_file && head -1 ~/hello-world >> ~/new_file

The AND operator (&&) will execute the second command only if, when executing the first command, SUCCEEDS

3 Suggest a new way to accomplish p.2, i.e. with new commands compared to p.2

less ~/hello-world | sort -r > ~/new_file

4 Solve the same task in a new way compared to p.2 and p.3, without using egrep (grep, fgrep), tail, head. Write the result into a new output file. Note: if some commands not shown in the course were used to accomplish p.2-4, please describe their logic.

tac ~/hello-world > ~/new_file

tac filename - This command is used to display information from a file in reverse order.

5 Concatenate the content of /etc/passwd, /etc/group, /etc/hosts, /etc/resolv.conf, then (every subtask below requires its own solution):

1. **Write the result into a file**
2. **Count the number of lines in this file**
3. **Without writing the result into a file, find there all lines containing “server”, then print them to the terminal**

cat /etc/passwd /etc/group /etc/hosts /etc/resolv.conf

5.1.cat /etc/passwd /etc/group /etc/hosts /etc/resolv.conf > ~test_cat

5.2 wc -l ~/test_cat

5.3 egrep 'server' ~/test_cat

6 In directory /apps/logs and all included subdirectories do the following:

6.1 Find all files containing “fail” (case-insensitive) in their content (only in their content, not in their names). Only file names should be printed to the terminal.

6.2 Find and print all lines containing “ERROR” (case-sensitive). These lines should be written to an output file. Then ensure this resulting file is not empty.

6.1 grep -ril "fail" /apps/logs

This command uses grep with the options:

-r: Recursively search through directories.

-i: Perform a case-insensitive match.

-l: List only the names of files with matching lines.

6.2 grep "ERROR" -r /apps/logs > output_file.txt

cat output_file.txt

7 Find all files in /etc (and all included subdirectories) containing your username. Only file names should be handled (not matching lines). Do the following with them - each subtask requires its own solution:

grep -rl 'ipapara' /etc/*

-r: Recursively search through directories.

-l: List only the names of files with matching lines.

7.1 Place the result and errors into 2 separate files

grep -rl 'ipapara' /etc/* > ~/file 2> ~/file_errors

After the execution of this command we can see that this file “file” contains only file names

```
[ipapara@c7-sandbox ~]$ cat file
/etc/group
/etc/group-
/etc/passwd
/etc/passwd-
/etc/subgid
/etc/subgid-
/etc/subuid
/etc/subuid-
```

All errors now is in this file file_errors

```
grep: /etc/lvm/archive: Permission denied
grep: /etc/openldap/certs/password: Permission denied
grep: /etc/pki/rsyslog: Permission denied
grep: /etc/pki/CA/private: Permission denied
grep: /etc/polkit-1/rules.d: Permission denied
grep: /etc/polkit-1/localauthority: Permission denied
grep: /etc/securetty: Permission denied
grep: /etc/security/opasswd: Permission denied
grep: /etc/selinux/targeted/semanage.trans.LOCK: Permission denied
grep: /etc/selinux/targeted/active: Permission denied
grep: /etc/selinux/targeted/semanage.read.LOCK: Permission denied
grep: /etc/selinux/final: Permission denied
grep: /etc/shadow: Permission denied
grep: /etc/shadow-: Permission denied
```

7.2 Place the result and errors into the same files, without using temporary files

grep -rl 'ipapara' /etc/* >~/file_all 2>&1

We can see that all info is in one file

```
grep: /etc/openldap/certs/password: Permission denied
/etc/passwd
/etc/passwd-
grep: /etc/pki/rsyslog: Permission denied
grep: /etc/pki/CA/private: Permission denied
grep: /etc/polkit-1/rules.d: Permission denied
grep: /etc/polkit-1/localauthority: Permission denied
grep: /etc/securetty: Permission denied
grep: /etc/security/opasswd: Permission denied
grep: /etc/selinux/targeted/semanage.trans.LOCK: Permission denied
grep: /etc/selinux/targeted/active: Permission denied
grep: /etc/selinux/targeted/semanage.read.LOCK: Permission denied
grep: /etc/selinux/final: Permission denied
grep: /etc/shadow: Permission denied
grep: /etc/shadow-: Permission denied
grep: /etc/squid/squid.conf: Permission denied
grep: /etc/ssh/sshd_config: Permission denied
grep: /etc/ssh/ssh_host_rsa_key: Permission denied
grep: /etc/ssh/ssh_host_ed25519_key: Permission denied
grep: /etc/ssh/ssh_host_ecdsa_key: Permission denied
/etc/subgid
/etc/subgid-
/etc/subuid
/etc/subuid-
grep: /etc/sudo.conf: Permission denied
```

7.3 Count the number of lines in p.a and p.b above

In case 7.1 for file without errors

```
[ipapara@c7-sandbox ~]$ wc -l  ~/file
8 /home/ipapara/file
```

In case 7.1 for file only with errors

```
[ipapara@c7-sandbox ~]$ wc -l ~/file_errors
51 /home/ipapara/file_errors
```

In case 7.2

```
[ipapara@c7-sandbox ~]$ grep -rl 'ipapara' /etc/*  >~/file_all  2>&1
[ipapara@c7-sandbox ~]$ wc -l ~/file_all
59 /home/ipapara/file_all
```

We can see that all is ok, because together they make 59 in sum

8 In directory /apps/logs and all included subdirectories find and print all lines from all files containing the following patterns: “error”, or “fail”, or “bug”. This search should be case-insensitive. Write the result into an output file. Notes: a) one line may contain more than 1 pattern b) the task can be solved via either “egrep -e” or regular expressions.

egrep -r -i -e "error|fail|bug" /apps/logs > ~/output.txt

-r: Recursively search through directories.

-i: Perform case-insensitive matching.

-e: Specify the pattern to search for, in this case, "error|fail|bug".

The | (pipe) character is a logical OR in regular expressions. So, the pattern "error|fail|bug" means "match lines that contain either 'error', 'fail', or 'bug'"**.**

We can check the result file with less ~/output.txt and find error , bug or fail words inside

```
/apps/logs/java-ftb.st00.log:Jun 25 06:58:27 c7-backend java-start-ftb.sh[12644]:
00xea0x030x03{0x0fbG0x010xd30x1a0x960xe2G0xc60x0f0xb60x8d0x040x93_0xdbug0x00S8@g-0
/apps/logs/java-ftb.st00.log:Jun 25 15:48:30 c7-backend java-start-ftb.sh[12644]:
eader
/apps/logs/java-ftb.st00.log:Jun 25 15:48:30 c7-backend java-start-ftb.sh[12644]:
```

9 Find all lines from /etc/passwd containing “/bin/bash”, but not containing “root”. Write the results into an output file.

grep "/bin/bash" /etc/passwd | grep -v "root" > ~/output_9.txt

The second grep command with the -v option inverts the match, excluding lines that contain "root" from the results.

```
vchapurin:x:2020:1001::/home/vchapurin:/bin/bash
imuslimova:x:2021:1001::/home/imuslimova:/bin/bash
vantropov:x:2023:1001::/home/vantropov:/bin/bash
nvasileva:x:2024:1001::/home/nvasileva:/bin/bash
eshiyanova:x:2025:1001::/home/eshiyanova:/bin/bash
aorlov:x:2026:1001::/home/aorlov:/bin/bash
dbatlukov:x:2027:1001::/home/dbatlukov:/bin/bash
nsklianova:x:2028:1001::/home/nsklianova:/bin/bash
egusev:x:2029:1001::/home/egusev:/bin/bash
nzharkov:x:2030:1001::/home/nzharkov:/bin/bash
nkovaleva:x:2031:1001::/home/nkovaleva:/bin/bash
slarina:x:2032:1001::/home/slarina:/bin/bash
agrebennikova:x:2033:1001::/home/agrebennikova:/bin/bash
ikalenikovich:x:2034:1001::/home/ikalenikovich:/bin/bash
akritskaya:x:2035:1001::/home/akritskaya:/bin/bash
```

We can open file . Command less output.txt and try to search root. There is no such word, because it is written : pattern not found

10 Find all lines from /etc/passwd containing “/home”. Write the results into another output file.

grep "/home" /etc/passwd > ~/output_home_10.txt

```
smironov:x:2101:1001::/home/smironov:/bin/bash
st14:x:2102:1001::/home/st14:/bin/bash
st15:x:2103:1001::/home/st15:/bin/bash
kmuntian:x:2104:1001::/home/kmuntian:/bin/bash
ookhapkina:x:2105:1001::/home/ookhapkina:/bin/bash
mcherepanina:x:2106:1001::/home/mcherepanina:/bin/bash
myakovenko:x:2107:1001::/home/myakovenko:/bin/bash
ipapara:x:2108:1001::/home/ipapara:/bin/bash
akaminskaya:x:2109:1001::/home/akaminskaya:/bin/bash
asinelnikov:x:2110:1001::/home/asinelnikov:/bin/bash
nlizunova:x:2111:1001::/home/nlizunova:/bin/bash
nkhaytovich:x:2112:1001::/home/nkhaytovich:/bin/bash
vbelov:x:2113:1001::/home/vbelov:/bin/bash
st2023711:x:2114:1001::/home/st2023711:/bin/bash
st2023712:x:2115:1001::/home/st2023712:/bin/bash
st2023713:x:2116:1001::/home/st2023713:/bin/bash
st2023714:x:2117:1001::/home/st2023714:/bin/bash
st2023715:x:2118:1001::/home/st2023715:/bin/bash
lurmazova:x:2119:1001::/home/lurmazova:/bin/bash
lkozinskaya:x:2002:1001::/home/lkozinskaya:/bin/bash
```

11 Compare files from p.9 and p.10 - at least, print their size and numbers of lines.

```
[ipapara@c7-sandbox ~]$ grep "/bin/bash" /etc/passwd | grep -v "root" > output_9.txt
[ipapara@c7-sandbox ~]$ ls -l output_9.txt
-rw-r--r--. 1 ipapara qa 5333 Dec  2 22:37 output_9.txt
```

```
ipapara@c7-sandbox ~]$ grep "/home" /etc/passwd > output_home_10.txt
ipapara@c7-sandbox ~]$ ls -l output_home_10.txt
rw-r--r--. 1 ipapara qa 5381 Dec  2 22:38 output_home_10.txt
```

```
[ipapara@c7-sandbox ~]$ wc -l output_9.txt
110 output_9.txt
```

```
[ipapara@c7-sandbox ~]$ wc -l output_home_10.txt
111 output_home_10.txt
```