

## Module 4 Linux & Bash Essentials (Task 4.9)

4.9.1 Проверить расположение командного интерпретатора bash и использовать этот путь в скриптах `#!/полный_путь_к_программе` опции\_программы

4.9.2 Проверяем, что bash запущен, используя специальную встроенную переменную.

```
bruh@wibob-X61:~$ echo $0
bash
bruh@wibob-X61:~$ which bash
/bin/bash
bruh@wibob-X61:~$ echo $SHELL
/bin/bash
bruh@wibob-X61:~$ echo $TERM
xterm-256color
bruh@wibob-X61:~$ echo $PS1
\[e]0;\u@h: \w\A)\${debian_chroot:+($debian_chroot)}\[033[01;32m]\u@\h\[033[00m]:\[033[01;34m]\w\[033[00m]\$
bruh@wibob-X61:~$ echo $PATH
/home/bruh/bin:/home/bruh/.local/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
bruh@wibob-X61:~$ ps -p $$
PID TTY      TIME CMD
  10  pts/0    00:00 bash
```

4.9.3 Создать директорию, где будут храниться ваши скрипты, например, `.../tmp/scripts`.

4.9.4 Создать сценарий, который выведет следующее сообщение «В моем домашнем каталоге <n> поддиректорий: <название>». n – количество, название – имя директории. Сделать этот скрипт исполняемой программой и запустить с помощью команд интерпретации в текущем командном интерпретаторе (это касается и всех последующих скриптов).

*## `home_sub.sh` (здесь и далее, где задание предполагает скрипт будут идти имена файлов скриптов лежащих на [github](#))*

```
~/.ssh (zsh)  %1  bruh@wibob-X61: ~/devops/scripts.d (ssh)  %2
bruh@wibob-X61:~/devops/scripts.d$ cat home_sub.sh
#!/bin/bash
printf "There are $(ls -lR ~ | grep '^d' | wc -l) subdirectories in my home directory: \n"
ls -lR ~ | awk '{if ($1 ~ /\^\/.*\.$/) cur_dir=$1; else if (NF>=9 && $1 ~ /\^d/) print substr(cur_dir, 1, length(cur_dir)-1) "/" $9; else;}'
bruh@wibob-X61:~/devops/scripts.d$ ./home_sub.sh
There are 219 subdirectories in my home directory:
/home/bruh/Desktop
/home/bruh/Documents
/home/bruh/Downloads
/home/bruh/LAB
/home/bruh/Music
/home/bruh/NFS
/home/bruh/Pictures
/home/bruh/Public
/home/bruh/Templates
/home/bruh/Videos
/home/bruh/devops
/home/bruh/git
/home/bruh/openssl
/home/bruh/test
/home/bruh/devops/1st_projects
/home/bruh/devops/2nd_project
/home/bruh/devops/scripts.d
/home/bruh/git/Documentation
/home/bruh/git/block-sha1
/home/bruh/git/builtin
/home/bruh/git/ci
/home/bruh/git/compat
/home/bruh/git/contrib
/home/bruh/git/ewah
/home/bruh/git/git-gui
/home/bruh/git/gitk-git
/home/bruh/git/gitweb
/home/bruh/git/mergetools
/home/bruh/git/negotiator
/home/bruh/git/perl
/home/bruh/git/po
/home/bruh/git/ppc
/home/bruh/git/refs
/home/bruh/git/sha1collisiondetection
```

4.9.5 Создать сценарий, который читает с экрана слово и выводит количество символов в этом слове.

*## wconline.sh с обрезанием пробелов и проваркой количества слов*

```

bruha@wibob-X61: ~/devops/scripts.d
~ (zsh) %1 bruha@wibob-X61: ~ (ssh) %2
bruha@wibob-X61:~/devops/scripts.d$ cat wconline.sh
#!/bin/bash
echo "Enter a word folowed by [Enter]"
read magicword
echo -n $magicword | sed 's:\s+:1' | awk '{if (NF>1) print "You enter more than 1 word! Try again."; else print length($1);}'
bruha@wibob-X61:~/devops/scripts.d$ ./wconline.sh
Enter a word folowed by [Enter]
Enter
5
bruha@wibob-X61:~/devops/scripts.d$ ./wconline.sh
Enter a word folowed by [Enter]
football
8
bruha@wibob-X61:~/devops/scripts.d$ ./wconline.sh
Enter a word folowed by [Enter]
Heavy Metal
You enter more than 1 word! Try again.
bruha@wibob-X61:~/devops/scripts.d$

```

4.9.6 Написать скрипт, который последовательно (с помощью позиционных параметров) принимает три аргумента a, b и c и выводит значение (a+b)/c в виде простой дроби.

*## fraction.sh (с комментариями) на скриншотах могут быть не все фрагменты, присутствующие в файле.*

```

int_patt='^[0-9]+$' # шаблон для определения целого

if [ "$#" -ne 3 ]; then # проверка, переданы ли 3 аргумента
    echo "Please enter three integer number"
else
    if [[ $1 =~ $int_patt && $2 =~ $int_patt && $3 =~ $int_patt ]]; then # проверка, аргументов на соответствие шаблону
        if [ "$3" -eq 0 ]; then # проверка деления на 0
            echo "Result: ERROR. Devision by Zero"
        else
            devident_ln=`echo -n "$1 + $2" | wc -c` #вычисляем длину делителя в символах отображения`
            devisor_ln=`echo -n "$3" | wc -c` # вычисляем длину делителя в символах
            if [ $devident_ln -gt $devisor_ln ]; then # определяем длину максимальной длины
                max_ln=$devident_ln
            else
                max_ln=$devisor_ln
            fi
            printf "%s\n" $(( ( $devident_ln + $max_ln ) / 2 )) "$1 + $2" # выводим отцентрировано выражение в числителе
            for ((i=0; i < $max_ln; i++)); do # в цикле рисуем дробь нужной длины
                echo -n "-"
            done
            echo -n " = "; echo "( $1 + $2 ) / $3" | bc -l # выводим в десятичном виде результат преобразования дроби
            printf "%s\n" $(( ( $devisor_ln + $max_ln ) / 2 )) "$3" # центрируем и выводим знаменатель дроби
        fi
    else
        echo "Sorry, positive integers only"
    fi
fi

```

```

bruha@wibob-X61: ~/devops/scripts
~.ssh (zsh) %1 ~ (zsh)
bruha@wibob-X61:~/devops/scripts.d$ ./fraction.sh 4 5 7
4 + 5
----- = 1.28571428571428571428
7
bruha@wibob-X61:~/devops/scripts.d$ ./fraction.sh 4 5 777777777776
4 + 5
----- = .00000000011571428571
77777777776
bruha@wibob-X61:~/devops/scripts.d$

```

```
vim (vim) ● 361 | bruha@wibob-X61: ~/devops/scripts.d (ssh)
```

```
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 55 32
55
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 55 675
675
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 33 33
Arguments are equal
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 33
Please enter two integer number
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 33 3 6
Please enter two integer number
bruha@wibob-X61:~/devops/scripts.d$ ./which_grater.sh 33 -3
Please, use positive integer number
bruha@wibob-X61:~/devops/scripts.d$
```

#### 4.9.8 Проверить, на какой позиции находится символ 'a' в path.

*# where\_letter.sh (для уточнения ввёл дополнительные условия: ищется первое вхождение буквы в слове которое передаётся в качестве аргумента, поиск идёт только в первом переданном аргументе, искомый символ задаётся в начале скрипта в переменной "letter")*

```
vim (vim)  %1  bruh@wibob-X61: ~/devops/scripts.d (ssh)

#!/bin/bash -
#=====
#
#       FILE: where_letter.sh
#
#       USAGE: ./where_letter.sh
#
#       DESCRIPTION:
#
#       OPTIONS: ---
#       REQUIREMENTS: ---
#       BUGS: ---
#       NOTES: ---
#       AUTHOR: Oleksandr Briukhanov (sysop), a.briukhanov@gmail.com
#       ORGANIZATION: wibob.net
#       CREATED: 02.05.20 02:13:17
#       REVISION: ---
#=====
set -o nounset                                # Treat unset variables as an error

letter=a
if [ "$#" -lt 1 ]; then                       # проверка, передан ли аргумент
    echo "Please use argument"
else
    for (( i=0; i<${#1}; i++ )); do
        if [ X${1:i:1} = X$letter ]; then
            echo $((i+1))
            break
        fi
    done
fi
```

```
vim (vim)  %1  bruh@wibob-X61: ~/devops/scripts.d (ssh)

bruh@wibob-X61:~/devops/scripts.d$ ./where_letter.sh path
2
bruh@wibob-X61:~/devops/scripts.d$ ./where_letter.sh
Please use argument
bruh@wibob-X61:~/devops/scripts.d$ ./where_letter.sh ddfghh path
bruh@wibob-X61:~/devops/scripts.d$ ./where_letter.sh "ddfghh path"
9
bruh@wibob-X61:~/devops/scripts.d$
```

#### 4.9.9 Удалить из date название часового пояса (EEST). Переставить день и месяц местами.

```
bruh@wibob-X61:~/devops/scripts.d$ date
субота, 2 травня 2020 08:11:53 +0300
bruh@wibob-X61:~/devops/scripts.d$ date | awk '{print $1, $3, $2, $4, $5}'
субота, травня 2 2020 08:11:56
bruh@wibob-X61:~/devops/scripts.d$
```

4.9.10 Вывести строку, содержащую предложение: "Я заходил в директорию <текущий каталог> в <время> используя идентификатор <UID>".

```
bru@wibob-X61:~/devops/scripts.d$ echo "Я заходил в директорию $(pwd) в $(ls -ld --time=atime `pwd` | awk '{print $8}') используя идентификатор $UID"
Я заходил в директорию /home/bru/devops/scripts.d в 08:00 используя идентификатор 1000
bru@wibob-X61:~/devops/scripts.d$
```

4.9.11 Вывести из \$PATH путь к последнему в списке каталогу (3 способа).

```
echo $PATH | sed -e 's/^.*\:\/\/'
echo $PATH | awk -F: '{print $NF}'
echo $PATH | rev | cut -d ':' -f1 | rev
```

```
bru@wibob-X61:~/devops/scripts.d$ echo $PATH | sed -e 's/^.*\:\/\/'
/snap/bin
bru@wibob-X61:~/devops/scripts.d$ echo $PATH | awk -F: '{print $NF}'
/snap/bin
bru@wibob-X61:~/devops/scripts.d$ echo $PATH | rev | cut -d ':' -f1 | rev
/snap/bin
```

4.9.12 Попытаться вывести значение переменной, не объявленной ранее и получить указанное вами сообщение об ошибке.

```
## var_error.sh
```

```
bru@wibob-X61:~/devops/scripts.d$ ./var_error.sh
./var_error.sh: line 22: compname: company name not set
bru@wibob-X61:~/devops/scripts.d$ ./var_error.sh EPAM
Company name is EPAM
bru@wibob-X61:~/devops/scripts.d$
```

4.9.13 В скрипте с помощью `wget` скачать документ с головной страницы сервера `mail.ru` и перенаправить полученный документ на вход `links` (`lynx`).

*## в логе команды `wget` находим имя сохраненного файла и вызываем браузер `lynx` в режиме дампа скачанной страницы (дальше можно парсить уже дамп, например)*

```
#!/bin/bash -
#=====
#
#   FILE: wget_lynx.sh
#
#   USAGE: ./wget_lynx.sh
#
# DESCRIPTION:
#
#   OPTIONS: ---
# REQUIREMENTS: ---
#   BUGS: ---
#   NOTES: ---
#   AUTHOR: Oleksandr Briukhanov (sysop), a.briukhanov@gmail.com
# ORGANIZATION: wibob.net
#   CREATED: 02.05.20 06:03:02
#   REVISION: ---
#=====

set -o nounset                # Treat unset variables as an error
wget http://adv-it.net -o wget_log.txt
path_link=$(cat wget_log.txt | grep Saving | cut -d ' ' -f 3 | tr -d '"')
lynx -dump file:///path_link
```