

Міністерство освіти і науки України
Одеський національний політехнічний університет
Інститут комп'ютерних систем
Кафедра інформаційних систем

Лабораторна робота №5
з дисципліни «Операційні Системи»

Тема: «Основи використання скриптової мови інтерпретатору оболонки
командного рядку»

Виконав:
ст. гр. АІ-204
Караульний Д. М

Перевірив:
Блажко О. А.

Одеса – 2021

Мета: придбання навичок автоматизації керування ОС з використанням скриптової мови інтерпретатора оболонки командного рядку.

Завдання для виконання:

1. Розробіть програму MyOSParam на мові BASH, яка виводить на екран лише окремі дані про параметри поточного стану ОС з віртуальної файлової системи procfs у відповідності із варіантом з таблиці.

Програма повинна:

- 1) отримувати назву параметру з командного рядка;
- 2) знаходити значення параметру у відповідному файлі
- 3) виводити на екран значення параметру та опис призначення параметру українською мовою.

| № команди | № учасника команди | 1-й параметр пам'яті | 2-й параметр пам'яті | П а р а м е т р п р о ц е с с е р у |
|-----------|--------------------|----------------------|----------------------|--|
| 4 | 2 | <i>Inactive</i> | <i>Bounce</i> | <i>Cache size</i> |

2. Розробіть програму на мові BASH, яка виконує дії у відповідності з варіантом, представленим у таблиці.

| № команди | № учасника команди | Назва файлу-скрипту | Обмеження на довжину назви | Обмеження на зміст назви |
|-----------|--------------------|----------------------------|----------------------------|--------------------------|
| 4 | 2 | <i>my_create_directory</i> | Не більше 28 | Не більше 3 цифр підряд |

Для всіх варіантів передбачається наступний опис кроків алгоритму роботи програми:

- 1) запропонувати ввести назву об'єкту, використовуючи українське запрошення;

2) перевірити відсутність в ОС об'єкту з такою назвою, використовуючи відповідні команди та файли;

3) якщо об'єкт присутній, тоді вивести на екран відповідне повідомлення про помилку та завершити роботу програми;

4) якщо назва об'єкту не відповідає вказаному обмеженню, тоді вивести на екран відповідне повідомлення про помилку та завершити роботу програми;

5) якщо назва об'єкту відповідає вказаному обмеженню, виконати відповідну команду.

У кроках алгоритму об'єктом може виступати файл або каталог, в залежності від варіанту завдання.

3. В попередній лабораторній роботі ви створили файл CSV-формату за пунктом 2.3

Розробіть програму на мові BASH, яка автоматично:

- читає зміст файлу з урахуванням табличної структури (рядки, стовпчики), використовуючи перенаправлення потоку;
- створює каталог з назвою, співпадаючою з назвою 3-го стовпчика таблиці файлу
- у створеному каталозі створює файли, назви яких співпадають зі значеннями 3-го стовпчика таблиці файлу

Хід роботи:

0. Копіюємо файл підсвітки синтаксису скриптів у свій домашній каталог користувача (рис. 0).

```
[karaulnij_daniil@vpsj3IeQ ~]$ cp /usr/share/nano/sh.nanorc ~/.nanorc  
[karaulnij_daniil@vpsj3IeQ ~]$
```

Рис. 0

1. Розробляють програму MyOSParam на мові BASH, яка виводить на екран лише окремі дані про параметри поточного стану ОС з віртуальної файлової системи procfs за заданих параметрах Inactive, Bounced – для виведення даних про оперативну пам'ять, cache size – для виведення інформації щодо процесору (код – рис. 1.1, результат виконання – рис 1.2).

```
karaulnij_daniil@vpsj3IeQ:~/OS-Lab5
GNU nano 2.3.1 File: MyOSParam.sh

#!/bin/bash

if (( $# == 0 ))
then
echo "Скрипт должен принимать один из параметров: Inactive, Bounce, cache_size."
else
if (( $# > 1 ))
then
echo "Скрипт не принимает более 1 параметра."
else
Key=0
case $* in
    "Inactive")
        Key=1
        echo "Вывод параметра памяти Inactive: "
        cat /proc/meminfo | grep "^Inactive";;
    "Bounce")
        Key=1
        echo "Вывод параметра памяти Bounce: "
        cat /proc/meminfo | grep "^Bounce";;
    "cache_size")
        Key=1
        echo "Вывод параметра процессора cahe size: "
        cat /proc/cpuinfo | grep "^processor\|^cahe size";;
esac

if (( Key == 0 ))
then
echo "Ошибка! Проверьте корректно переданного параметра"
fi

fi
fi
```

Рис. 1.1

```
karaulnij_daniil@vpsj3IeQ:~/OS-Lab5
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh
Скрипт должен принимать один из параметров: Inactive, Bounce, cache_size.
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh Inactive Bounce
Скрипт не принимает более 1 параметра.
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh Inactive
Вывод параметра памяти Inactive:
Inactive:          730420 kB
Inactive(anon):    375656 kB
Inactive(file):    354764 kB
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh Bounce
Вывод параметра памяти Bounce:
Bounce:            0 kB
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh cache_size
Вывод параметра процессора cahe size:
processor           : 0
processor           : 1
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ sh MyOSParam.sh ryui
Ошибка! Проверьте корректно переданного параметра
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$
```

Рис. 1.2

2. Розробляють програму my_create_directory на мові BASH, яка виконує видалення файлу із обмеженням не більше 28ти символів на довжину назви та не більше 3ьох цифр підряд (код – рис 2.1, результат виконання – рис 2.2).

```
karaulnij_daniil@vpsj3leQ:~/OS-Lab5
GNU nano 2.3.1 File: my_create_directory.sh

#!/bin/bash

while [[ "$directory_name" = "" ]]
do
    echo -n "Введите название директории: "
    read directory_name

    if [[ -n "directory_name" ]]
    then

        if [[ -d "$directory_name" ]]
        then
            echo "Директория с таким названием уже существует!"
        else
            if [[ ${#directory_name} -gt 28 ]]
            then
                echo "Название директории не может содержать более 28 символов"
            else
                if [[ $directory_name =~ [0-9]{4,} ]]
                then
                    echo "Название директории не может содержать более 3 цифр подряд!"
                else
                    mkdir $directory_name
                    echo "Директория $directory_name успешно создана!"
                fi
            fi
        fi
    fi
done
```

Рис 2.1

```
karaulnij_daniil@vpsj3leQ:~/OS-Lab5
[karaulnij_daniil@vpsj3leQ OS-Lab5]$ sh my_create_directory.sh
Введите название директории: testtesttesttesttesttesttesttest
Название директории не может содержать более 28 символов
[karaulnij_daniil@vpsj3leQ OS-Lab5]$ sh my_create_directory.sh
Введите название директории: test12345
Название директории не может содержать более 3 цифр подряд!
[karaulnij_daniil@vpsj3leQ OS-Lab5]$ sh my_create_directory.sh
Введите название директории: test123
Директория test123 успешно создана!
[karaulnij_daniil@vpsj3leQ OS-Lab5]$ sh my_create_directory.sh
Введите название директории: test123
Директория с таким названием уже существует!
[karaulnij_daniil@vpsj3leQ OS-Lab5]$ ls
KaraulniyDaniil.csv my_create_directory.sh MyOSParam.sh my_read_csv.sh test123 test.csv
[karaulnij_daniil@vpsj3leQ OS-Lab5]$
```

Рис 2.2

3. Розробляють програму на мові BASH, яка автоматично читає зміст файлу з урахуванням табличної структури (рядки, стовпчики) використовуючи перенаправлення потоку; створює каталог з назвою, співпадаючою з назвою 3-го стовпчика таблиці файлу у створеному каталозі

створює файли, назви яких співпадають зі значеннями 3-го стовпчика таблиці файлу (код – рис. 3.1, 3.2, результат виконання – рис 3.3).

```
karaulnij_daniil@vpsj3leQ:~/OS-Lab5
GNU nano 2.3.1 File: my_read_csv.sh

#!/bin/bash

while [[ "$file_name" = "" ]]
do
    echo -n "Введите название файла: "
    read file_name

    if [[ -n "$file_name" ]]
    then
        if [[ -f "$file_name" ]]
        then
            if [[ $file_name =~ .csv ]]
            then
                if (( $(head -n 1 $file_name | awk -F, '{print NF}' ) >= 3 ))
                then
                    line_num=0
                    files_count=0

                    while read line
                    do
                        if (( line_num == 0 ))
                        then
                            dir=$(echo $line | cut -d, -f3 | sed 's/ /_/g' | sed 's/[^a-zA-Za-zA-Я0-9]//g')
                            echo "Создан каталог $dir."
                            mkdir $dir
                            pushd $dir
                            line_num=1
                        else
                            file=$(echo $line | cut -d, -f3 | sed 's/ /_/g' | sed 's/[^a-zA-Za-zA-Я0-9]//g')
                            if [[ -n "$file" ]]
                            then
                                if [[ -f "$file" ]]
                                then
                                    echo "Файл $file уже существует."
                                else
                                    touch $file
                                    echo "Создание файла $dir/$file"
                                    (( files_count++ ))
                                fi
                            fi
                        fi
                    done < $file_name

                    echo "Создан 1 каталог и $files_count файл(ов)."
```

Рис. 3.1

```
            fi
        done < $file_name

        echo "Создан 1 каталог и $files_count файл(ов)."
```

```
    else
        echo "Файл содержит меньше, чем 3 столбца."
    fi
    echo "Файл должен иметь расширение \".csv\"."
fi
echo "Файл $file_name не существует."
fi
done
```

Рис. 3.2

```

Создание файла Усьороооі6/1154
Создание файла Усьороооі6/1154
Создан 1 каталог и 34 файл(ов).
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ ls
KaraulniyDaniil.csv my_create_directory.sh MyOSParam.sh my_read_csv.sh test123 test.csv Усьороооі6
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ less Усьороооі6

[1]+  Stopped                  less Усьороооі6
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ ls Усьороооі6
10985 13123 1380 16246 1764 1982 2218 2302 2607 2865 3089 3282 3842 4308 5034 5474 6513
1154 1359 14556 1707 17659 1989 2299 23059 273720 2997 3166 3444 4149 4423 5057 5504 9242
[karaulnij_daniil@vpsj3IeQ OS-Lab5]$ █

```

Рис. 3.3

Висновки: в ході виконання лабораторної роботи, були придбані навички автоматизації керування ОС з використанням скриптової мови інтерпретатору оболонки командного рядку. Найскладнішою частиною було налаштування перевірок назв файлів та їх інтерпретації для 3го завдання.