

Лабораторная работа 1

Задание 1

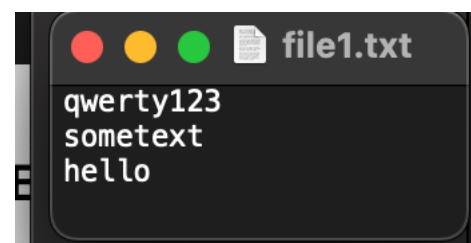
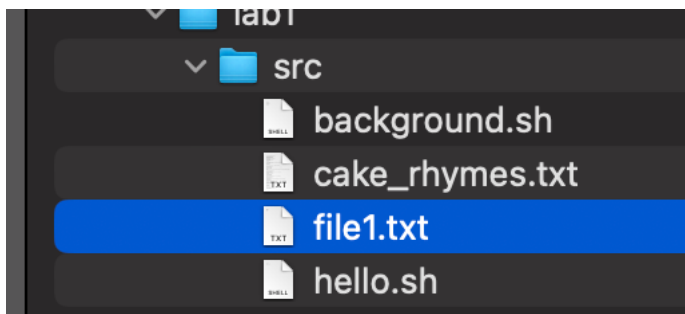
Запустите скрипт `background.sh` в фоновом режиме.

```
[veselkova@MacBook-Pro-Viktoria src % ./background.sh &
[1] 2440
veselkova@MacBook-Pro-Viktoria src % 1 sec
2 sec
3 sec
4 sec
5 sec
6 sec
7 sec
8 sec
9 sec
10 sec
Done!

[1] + done      ./background.sh
veselkova@MacBook-Pro-Viktoria src %
```

Создайте текстовый файл и выведите его содержимое на экран терминала.

```
[veselkova@MacBook-Pro-Viktoria src % cat>file1.txt
qwerty123
sometext
hello^D
veselkova@MacBook-Pro-Viktoria src %
```



Вывести

```
[veselkova@MacBook-Pro-Viktoria src % cat file1.txt
qwerty123
sometext
hello
veselkova@MacBook-Pro-Viktoria src %
```

- Посчитайте количество символов в файле.

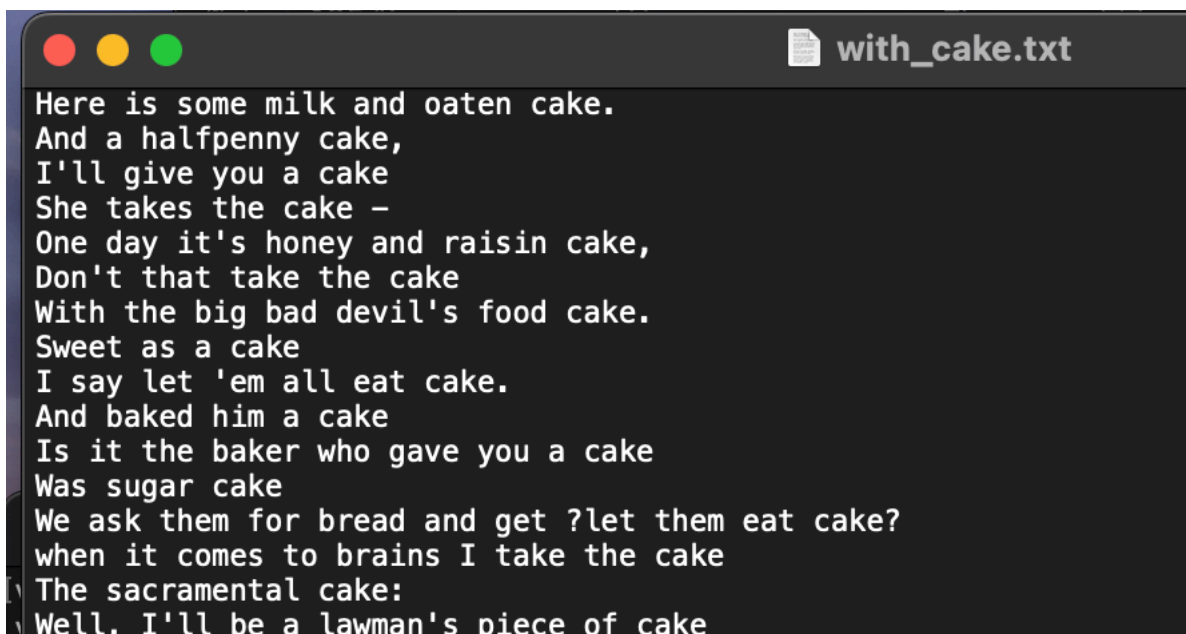
```
veselkova@MacBook-Pro-Viktoriasrc % wc -c file1.txt
25 file1.txt
veselkova@MacBook-Pro-Viktoriasrc %
```

Очистим консоль: `clear`

Задание 2

С помощью команды `grep`, используя `pipe` и редирект (`>`), запишите в файл `with_cake.txt` все строки из файла `cake_rhymes.txt`, в которых есть слово `cake`.

```
src % grep "cake" cake_rhymes.txt | > with_cake.txt
src %
```



```
with_cake.txt
Here is some milk and oaten cake.
And a halfpenny cake,
I'll give you a cake
She takes the cake -
One day it's honey and raisin cake,
Don't that take the cake
With the big bad devil's food cake.
Sweet as a cake
I say let 'em all eat cake.
And baked him a cake
Is it the baker who gave you a cake
Was sugar cake
We ask them for bread and get ?let them eat cake?
when it comes to brains I take the cake
The sacramental cake:
Well, I'll be a lawman's piece of cake
```

Сделайте команду `rm` "тихой", используя редирект в `/dev/null`.

```
rm -f file1 2>/dev/null
```

Задание 3

Сделайте файл `hello.sh` исполняемым, выполните его.

+x - это опция команды `chmod`, которая добавляет (плюс) право выполнения (x) к файлу.

```
veselkova@MacBook-Pro-Viktoria src % chmod +x hello.sh
```

```
veselkova@MacBook-Pro-Viktoria src % ./hello.sh
```

```
Hello, world!
```

Напишите `bash` скрипт, который выводит текущий путь, текущую дату и время, а также содержимое переменной окружения `PATH`.

Можно сразу создать файл например `script1.sh` и в него записать скрипт

```
cat > script1.sh
[veselkova@MacBook-Pro-Viktoria src % cat > script1.sh
#!/bin/bash

# Вывод текущего пути
echo "Текущий путь: $PWD"

# Вывод текущей даты и времени
current_date_time=$(date)
echo "Текущая дата и время: $current_date_time"

# Вывод содержимого переменной окружения PATH
echo "Переменная окружения PATH: $PATH"
veselkova@MacBook-Pro-Viktoria src %
```

Вывод:

```
veselkova@MacBook-Pro-Viktoria src % ./script1.sh
zsh: permission denied: ./script1.sh
```

Надо
разрешить и потом запускать

```

veselkova@MacBook-Pro-Viktoria src % chmod +x script1.sh
veselkova@MacBook-Pro-Viktoria src % ./script1.sh
Текущий путь: /Users/veselkova/Documents/Учёба/3 курс/операционные системы и
сети/lab1/src
Текущая дата и время: четверг, 21 сентября 2023 г. 14:41:29 (MSK)
Переменная окружения PATH: /Users/veselkova/.nvm/versions/node/v17.9.1/bin:/o
pt/homebrew/bin:/opt/homebrew/sbin:/Library/Frameworks/Python.framework/Versi
ons/3.9/bin:/usr/local/bin:/System/Cryptexes/App/usr/bin:/usr/bin:/bin:/usr/s
bin:/sbin:/usr/local/share/dotnet:~/dotnet/tools:/Library/Apple/usr/bin:/Lib
rary/Frameworks/Mono.framework/Commands:/Applications/Wiresh
ark.app/Contents/MacOS:/var/run/com.apple.security.cryptexd/codex.system/boot
strap/usr/local/bin:/var/run/com.apple.security.cryptexd/codex.system/bootstr
ap/usr/bin:/var/run/com.apple.security.cryptexd/codex.system/bootstrap/usr/ap
pleinternal/bin
veselkova@MacBook-Pro-Viktoria src %

```

Задание 4

Напишите скрипт `average.sh`, который выводит количество и среднее арифметическое его входных аргументов.

```

[veselkova@MacBook-Pro-Viktoria src % cat > average.sh
#!/bin/bash
# Объявляем переменные для суммы и количества аргументов
sum=0
count=0
# Проходим по всем аргументам и вычисляем сумму
for arg in "$@"; do
    sum=$((sum + arg))
    count=$((count + 1))
done
# Вычисляем среднее арифметическое
average=$(bc -l <<< "scale=2; $sum / $count")

# Выводим результат
echo "Количество аргументов: $count"
echo "Среднее арифметическое: $average"
[veselkova@MacBook-Pro-Viktoria src % chmod x+ average.sh
chmod: Invalid file mode: x+
[veselkova@MacBook-Pro-Viktoria src % chmod +x average.sh
veselkova@MacBook-Pro-Viktoria src %

```

Вывод

```

[veselkova@MacBook-Pro-Viktoria src % ./average.sh 2 3 4 5
Количество аргументов: 4
Среднее арифметическое: 3.50
veselkova@MacBook-Pro-Viktoria src %

```


С помощью `bash` и `dev/random` создайте файл `numbers.txt` из 150 рандомных чисел.

```
[veselkova@MacBook-Pro-Viktoria src % cat > random.sh
#!/bin/bash

# Определите имя файла
output_file="numbers.txt"

# Удаляем файл, если он уже существует
rm -f "$output_file"

# Генерируем 150 рандомных чисел и записываем их в файл
for i in {1..150}; do
    random_number=$(od -An -N2 -i /dev/random | awk '{print $1}')
    echo "$random_number" >> "$output_file"
done

# Выводим сообщение об успешном завершении
echo "Создан файл $output_file с 150 рандомными числами."
[veselkova@MacBook-Pro-Viktoria src % chmod +x random.sh
[veselkova@MacBook-Pro-Viktoria src % ./random.sh
Создан файл numbers.txt с 150 рандомными числами.
[veselkova@MacBook-Pro-Viktoria src % cat numbers.txt
31297
48037
23417
52407
27956
7883
40873
37836
54742
12267
```

"Скормите" скрипту `average.sh` значения из файла `numbers.txt`.

Надо чуток дописать скрипт `average.sh`

```
[veselkova@MacBook-Pro-Viktoria src % cat > average2.sh
#!/bin/bash
#можно прочитать значения из файла numbers.txt, если он существует
if [ -f "numbers.txt" ]; then
    numbers=$(cat numbers.txt)
fi

# Проверяем, что есть хотя бы одно число
if [ ${#numbers[@]} -eq 0 ]; then
    echo "No numbers provided or found in numbers.txt."
    exit 1
fi

# Объявляем переменные для суммы и количества аргументов
sum=0
count=0

# Проходим по всем аргументам (или числам из файла) и вычисляем сумму
for arg in "${numbers[@]"; do
    sum=$((sum + arg))
    count=$((count + 1))
done

# Вычисляем среднее арифметическое
average=$(bc -l <<< "scale=2; $sum / $count")

# Выводим результат
echo "Количество аргументов (или чисел из файла): $count"
echo "Среднее арифметическое: $average"
[veselkova@MacBook-Pro-Viktoria src % chmod +x average2.sh
[veselkova@MacBook-Pro-Viktoria src % ./average2.sh
Количество аргументов (или чисел из файла): 150
Среднее арифметическое: 31258.11
veselkova@MacBook-Pro-Viktoria src % █
```