

Отчёт по лабораторной работе №12

Простейший вариант

Ахмад Мд Шешир

Содержание

1	Цель работы	5
2	Задания	6
3	Выполнение лабораторной работы	7
3.1	Задача 1	7
3.2	Задача 2	9
3.3	Задача 3	10
3.4	Задача 4	11
4	Выводы	13

Список иллюстраций

3.1	Набераю текст	8
3.2	файл	11
3.3	программа	12
3.4	исполняемый файл	12
3.5	проверка	12

Список таблиц

1 Цель работы

Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы.

2 Задания

1. Написать скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в вашем домашнем каталоге. При этом файл должен архивироваться одним из архиваторов на выбор zip, bzip2 или tar

2. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять. Например, скрипт может последовательно распечатывать значения всех переданных аргументов.

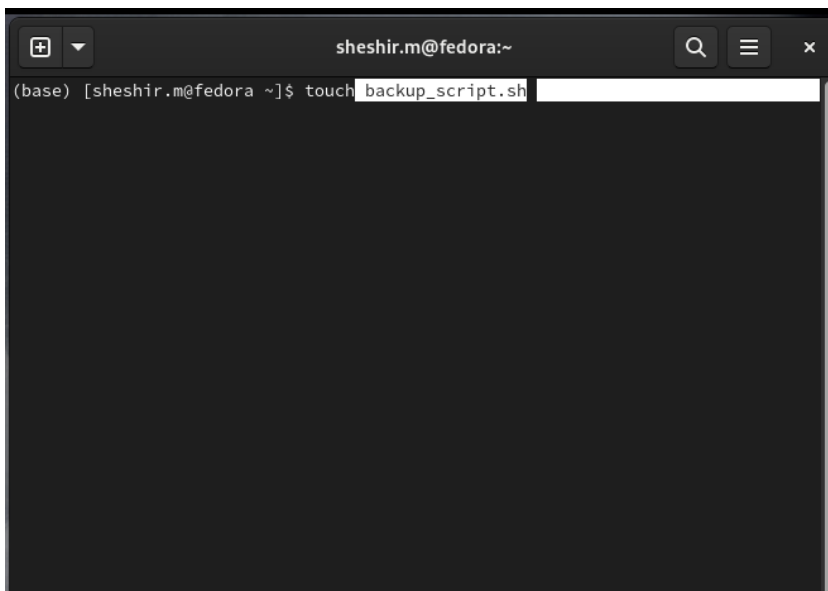
3. Написать командный файл — аналог команды ls (без использования самой этой команды и команды dir). Требуется, чтобы он выдавал информацию о нужном каталоге и выводил информацию о возможностях доступа к файлам этого каталога

4. Написать командный файл, который получает в качестве аргумента командной строки формат файла (.txt, .doc, .jpg, .pdf и т.д.) и вычисляет количество таких файлов в указанной директории. Путь к директории также передаётся в виде аргумента командной строки

3 Выполнение лабораторной работы

3.1 Задача 1

1. Создаю файл backup_script.sh где будет сам скрипт (рис. ??).



2. Набираю программу где создаю переменные с путями для резервной копии . (рис. 3.1).

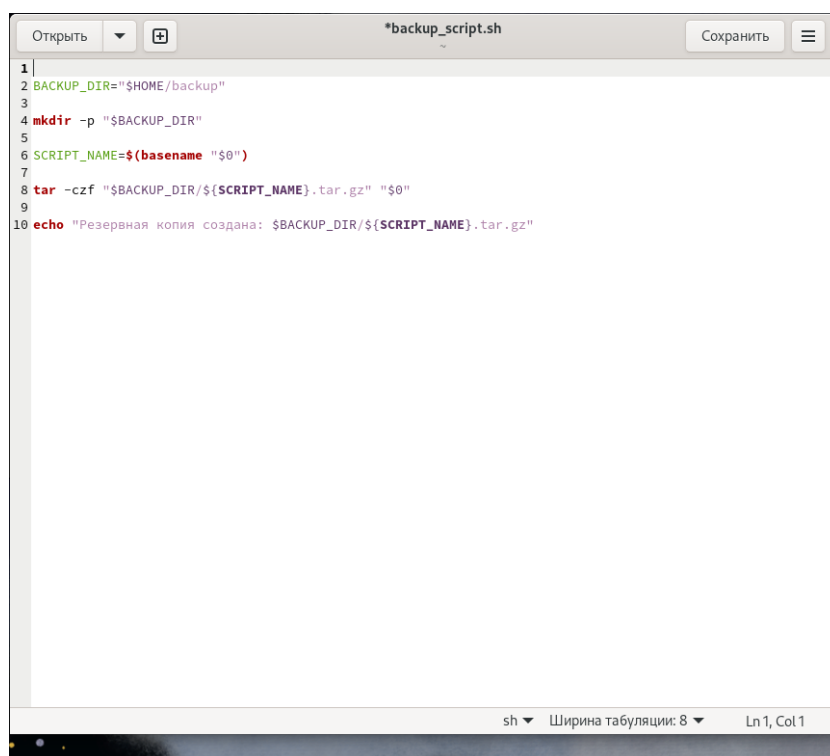


Рис. 3.1: Набираю текст

3. Делаю файл исполняемым, выполнив команду в терминале `chmod` (рис. ??).

```

(base) [sheshir.m@fedora ~]$ chmod +x backup_script.sh
(base) [sheshir.m@fedora ~]$

```

4. Запускаю исполняемый файл и проверяю существует ли архив (рис. ??).

```

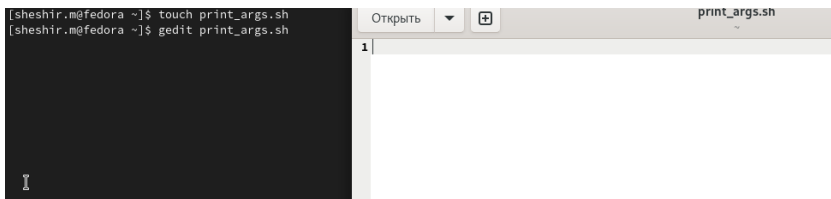
(base) [sheshir.m@fedora ~]$ bash backup_script.sh
Резервная копия создана: /home/sheshir.m/backup/backup_script.sh.tar.gz
(base) [sheshir.m@fedora ~]$

(base) [sheshir.m@fedora ~]$ ls /home/sheshir.m/backup/backup_script.sh.tar.gz
/home/sheshir.m/backup/backup_script.sh.tar.gz
(base) [sheshir.m@fedora ~]$

```


3.2 Задача 2

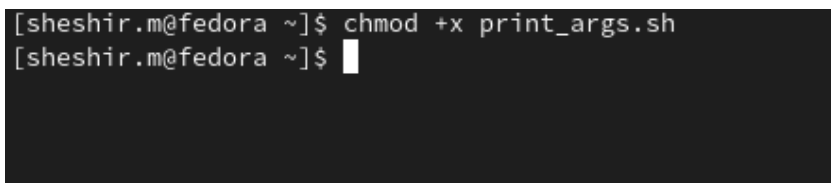
1. Открываю текстовый редактор и создайте новый файл print_args.sh (рис. ??).



2. Набираю программу которая будет сначала проверять переданы ли аргументы, а если переданы будет их перебирать с помощью цикла `for` и выводить (рис. ??).



3. Делаю файл исполняемым, выполнив команду в терминале `chmod` .(рис. ??).

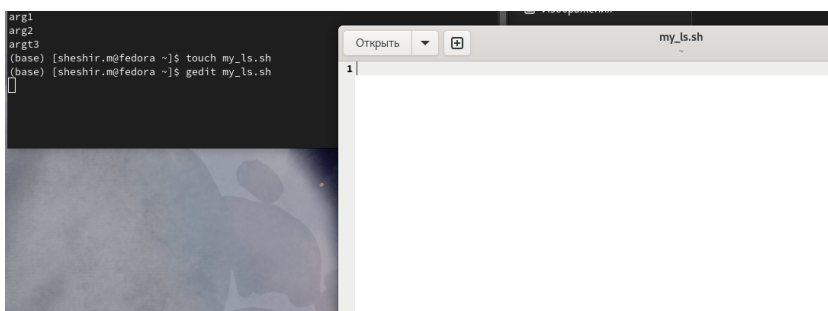


4. Запускаю исполняемый файл и проверяю работу (рис. ??).

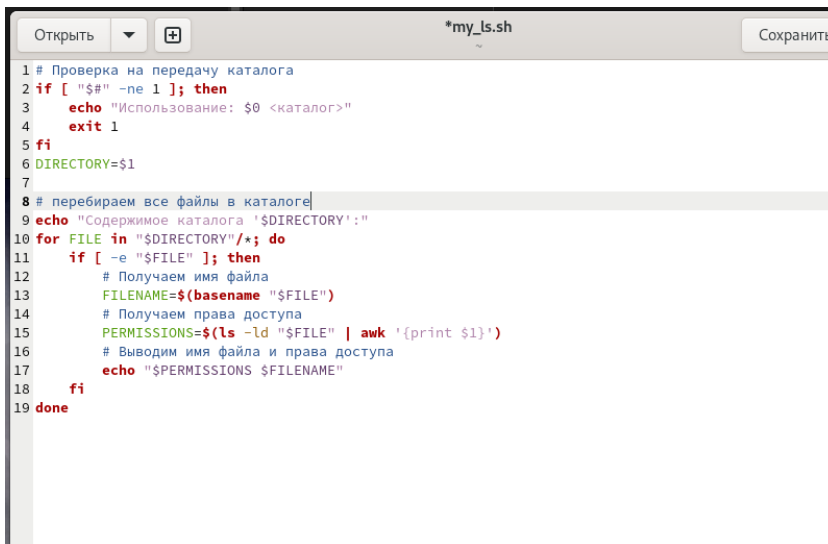
```
(base) [sheshir.m@fedora ~]$ ./print_args.sh arg1 arg2 argt3
Переданные аргументы:
arg1
arg2
argt3
(base) [sheshir.m@fedora ~]$
```

3.3 Задача 3

1. Открываю редактор gedit и создаю новый файл my_ls.sh (рис. ??).



2. Пишу программу которая будет проверять существование переданного каталога, и вывода всего его содержимого. (рис. ??).



3. Делаю файл исполняемым, выполнив команду в терминале chmod (рис. ??).

```
(base) [sheshir.m@fedora ~]$ chmod +x my_ls.sh
(base) [sheshir.m@fedora ~]$
```

4. Запускаю исполняемый файл, передаю ей в качестве аргумента путь до нужной директории. Проверяю работу (рис. ??).

```
(base) [sheshir.m@fedora ~]$ ./my_ls.sh /home/sheshir.m/lab7
Содержимое каталога '/home/sheshir.m/lab7':
-rw-r--r--. abc1
-rw-r--r--. australia
-rw-r--r--. dis.cpp
-rw-r--r--. equipment.h
-rw-rw-r--. feathers
-rw-r--r--. lab07.txt
-rwx---r--. my_os
-rw-r--r--. piu.py
-rwx-----. play
-rw-r--r--. text.txt
(base) [sheshir.m@fedora ~]$
```

3.4 Задача 4

1. Открываю редактор gedit и создаю новый файл count_files.sh (рис. 3.2).

```
(base) [sheshir.m@fedora ~]$ touch count_files.sh
(base) [sheshir.m@fedora ~]$
```

Рис. 3.2: файл

2. Пишу программу которая будет проверять существование переданного каталога, и вывода всего его содержимого. (рис. 3.3).



Рис. 3.3: программа

3. Делаю файл исполняемым, выполнив команду в терминале chmod (рис. 3.4).

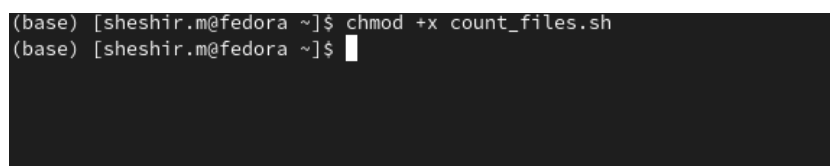


Рис. 3.4: исполняемый файл

4. Запускаю исполняемый файл, передаю ей в качестве аргумента путь до нужной директории и формал файла (png, txt doc и тд). Проверяю работу (рис. 3.5).

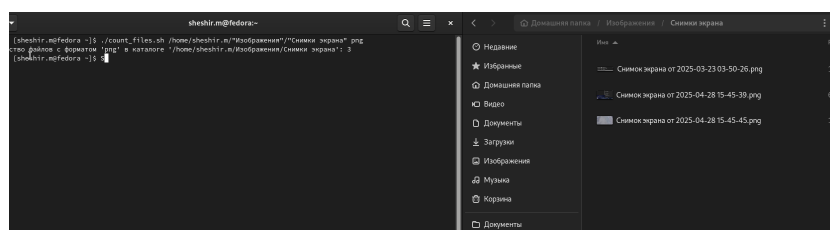


Рис. 3.5: проверка

4 Выводы

В ходе работы я познакомился с основами программирования в оболочке ОС UNIX/Linux. Научился писать небольшие командные файлы.