

Лабораторная работа №12

Презентация

Андрюшин Н. С.

12 марта 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Андрюшин Никита Сергеевич
- Студент
- Российский университет дружбы народов
- 1132231848@pfur.ru

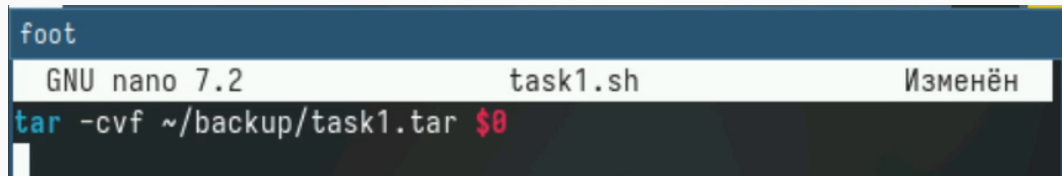
Изучить основы программирования в оболочке ОС UNIX/Linux. Научиться писать небольшие командные файлы

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. Способ использования команд архивации необходимо узнать, изучив справку.
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.

Код первой программы

Напишем код первой программы



The image shows a terminal window with a dark blue title bar containing the text 'foot'. Below the title bar is a light green header bar with the text 'GNU nano 7.2' on the left, 'task1.sh' in the center, and 'Изменён' on the right. The main area of the terminal is black and contains the command 'tar -cvf ~/backup/task1.tar \$0' in a light blue font. A red cursor is positioned at the end of the command.

```
foot
GNU nano 7.2          task1.sh          Изменён
tar -cvf ~/backup/task1.tar $0
```

Рис. 1: Код первой программы

Проверка работы первой программы

И проверим работу

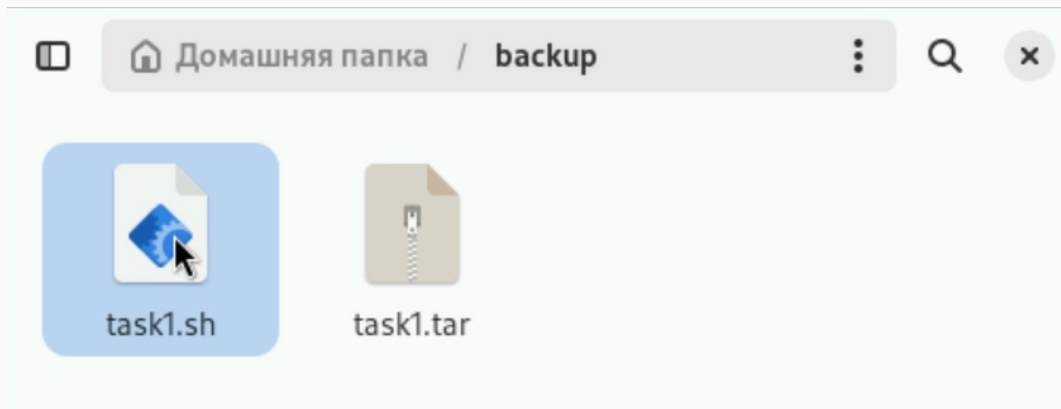
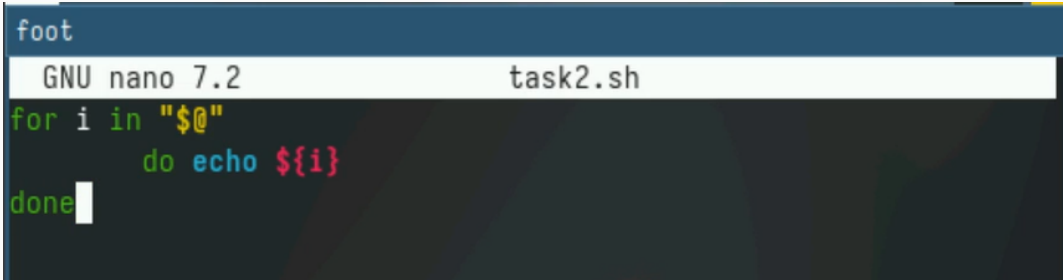


Рис. 2: Проверка работы первой программы

Код второй программы

Напишем код второй программы

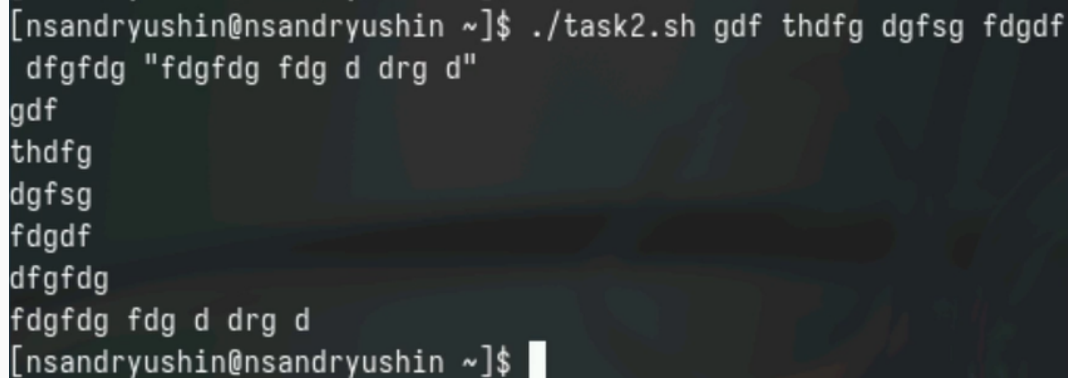


```
foot
GNU nano 7.2 task2.sh
for i in "$@"
do echo ${i}
done
```

Рис. 3: Код второй программы

Проверка работы второй программы

И проверим работу

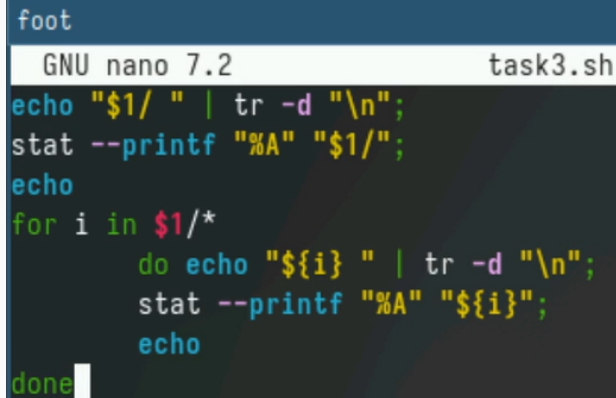


```
[nsandryushin@nsandryushin ~]$ ./task2.sh gdf thdfg dgfsg fdgdf  
dfgfdg "fdgfdg fdg d drg d"  
gdf  
thdfg  
dgfsg  
fdgdf  
dfgfdg  
fdgfdg fdg d drg d  
[nsandryushin@nsandryushin ~]$
```

Рис. 4: Проверка работы второй программы

Код третьей программы

Напишем код третьей программы



```
foot
GNU nano 7.2 task3.sh
echo "$1/ " | tr -d "\n";
stat --printf "%A" "$1/";
echo
for i in $1/*
do echo "${i} " | tr -d "\n";
stat --printf "%A" "${i}";
echo
done
```

Рис. 5: Код третьей программы

Проверка работы третьей программы

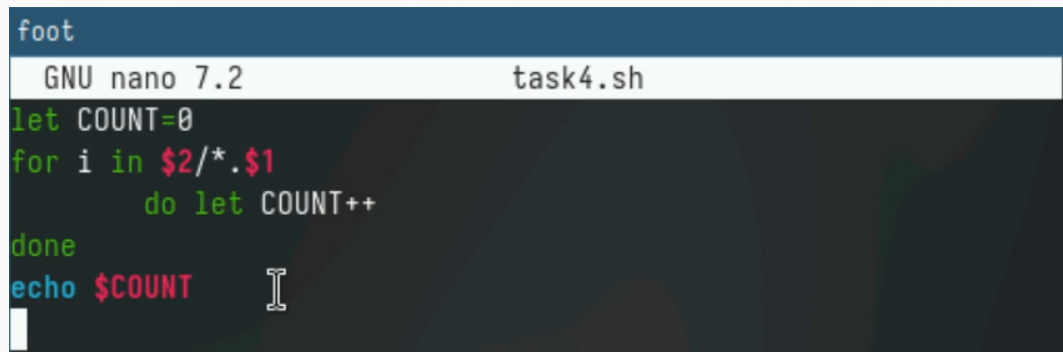
И проверим работу

```
[nsandryushin@nsandryushin ~]$ nano task3.sh
[nsandryushin@nsandryushin ~]$ ./task3.sh ~
/home/nsandryushin/ drwx-----
/home/nsandryushin/#1# -rw-r--r--
/home/nsandryushin/#2# -rw-r--r--
/home/nsandryushin/#3# -rw-r--r--
/home/nsandryushin/#4# -rw-r--r--
/home/nsandryushin/abc1 -rw-rw-r--
/home/nsandryushin/australia drwxr--r--
/home/nsandryushin/backup drwxr-xr-x
/home/nsandryushin/conf.txt -rw-r--r--
/home/nsandryushin/Downloads drwxr-xr-x
/home/nsandryushin/feathers -rw-rw-r--
/home/nsandryushin/file.txt -rw-r--r--
/home/nsandryushin/git-extended drwxr-xr-x
/home/nsandryushin/kconf.txt drwxr-xr-x
/home/nsandryushin/#lab07.sh# -rw-r--r--
/home/nsandryushin/lab07.sh -rw-r--r--
/home/nsandryushin/Lab1 drwxrwx---
```

Рис. 6: Проверка работы третьей программы

Код четвертой программы

Напишем код четвертой программы

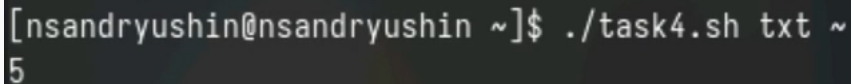


```
foot
GNU nano 7.2 task4.sh
let COUNT=0
for i in $2/*. $1
do let COUNT++
done
echo $COUNT
```

Рис. 7: Код четвертой программы

Проверка работы четвертой программы

И проверим работу

A terminal window with a dark background. The prompt is [nsandryushin@nsandryushin ~]\$ and the command being executed is ./task4.sh txt ~. The output of the command is the number 5.

```
[nsandryushin@nsandryushin ~]$ ./task4.sh txt ~  
5
```

Рис. 8: Проверка работы четвертой программы

В результате выполнения лабораторной работы были получены навыки написания скриптовых файлов