

# Лабораторная работа №10

Программирование в командном процессоре ОС UNIX. Командные файлы

---

Шуплецов А. А.

15 апреля 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Шуплецов Александр Андреевич
- студент ФФМиЕН
- Российский университет дружбы народов
- <https://github.com/winnralex>

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



## Изучим справку tar.

```
aashuplecov@aashuplecov:~$ man tar
TAR(1)                                GNU TAR Manual                                TAR(1)

NAME
   tar - an archiving utility

SYNOPSIS
   Traditional usage
   tar (A|c|d|r|t|u|X)[enskiWompambia]JzzhPlrvwo [ARG...]

   UNIX-style usage
   tar -A [OPTIONS] ARCHIVE ARCHIVE

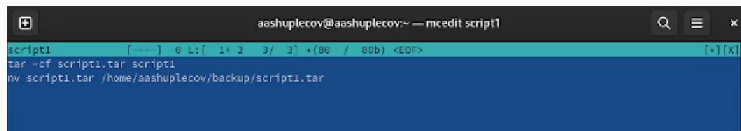
   tar -c [-f ARCHIVE] [OPTIONS] [FILE...]
   tar -d [-f ARCHIVE] [OPTIONS] [FILE...]
   tar -t [-f ARCHIVE] [OPTIONS] [MEMBER...]
   tar -r [-f ARCHIVE] [OPTIONS] [FILE...]
   tar -u [-f ARCHIVE] [OPTIONS] [FILE...]
   tar -x [-f ARCHIVE] [OPTIONS] [MEMBER...]

   GNU-style usage
   tar (--catenate|--concatenate) [OPTIONS] ARCHIVE ARCHIVE

   tar --create [--file ARCHIVE] [OPTIONS] [FILE...]
   tar (--diff|--compare) [--file ARCHIVE] [OPTIONS] [FILE...]
   tar --delete [--file ARCHIVE] [OPTIONS] [MEMBER...]
   tar --append [-f ARCHIVE] [OPTIONS] [FILE...]
   tar --list [-f ARCHIVE] [OPTIONS] [MEMBER...]
   tar --test-label [--file ARCHIVE] [OPTIONS] [LABEL...]
   tar --update [--file ARCHIVE] [OPTIONS] [FILE...]
   tar --update [-f ARCHIVE] [OPTIONS] [FILE...]
   tar (--extract|--get) [-f ARCHIVE] [OPTIONS] [MEMBER...]

NOTE
   This manpage is a short description of GNU tar. For a detailed discussion, including examples and usage
   Manual page tar(1) line 1 (press h for help or q to quit)
```

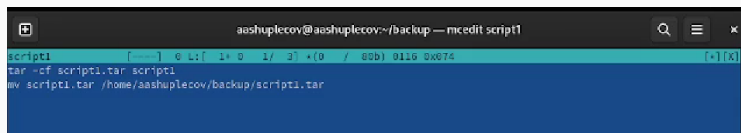
Напишем скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в нашем домашнем каталоге.

A screenshot of a terminal window with a dark background. The title bar at the top reads 'aashuplecov@aashuplecov:~ — mcedit script1'. The terminal content shows a script named 'script1' with two lines: 'tar -cf script1.tar script1' and 'mv script1.tar /home/aashuplecov/backup/script1.tar'. The first line is highlighted in light blue. The terminal has standard window controls (search, menu, close) in the top right corner.

```
script1 [---] 6 L: [ 1+ 2 3/ 3] *(80 / 80b) <EOF> [x] [X]  
tar -cf script1.tar script1  
mv script1.tar /home/aashuplecov/backup/script1.tar
```

Рис. 2: текст скрипта, делающий резервную копию

Убедимся, что скрипт, который при запуске будет делать резервную копию самого себя (то есть файла, в котором содержится его исходный код) в другую директорию backup в нашем домашнем каталоге, работает.

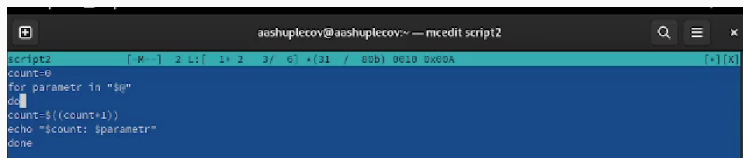


```
aashuplecov@aashuplecov:~/backup — mcedit script1
script1 [-----] 0 L: [ 1* 0 1/ 3] *{0 / 80b) 0116 0x074 [*)(X]
tar -cf script1.tar script1
mv script1.tar /home/aashuplecov/backup/script1.tar
```

Рис. 3: проверка скрипта, делающего резервную копию



Напишем пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять.



```
script2 [~] 2 L: 1 2 3 6 + (31 / 00b) 0010 0x00A [~] [X]
count=0
for parametr in "$@"
do
count=$((count+1))
echo "$count: $parametr"
done
```

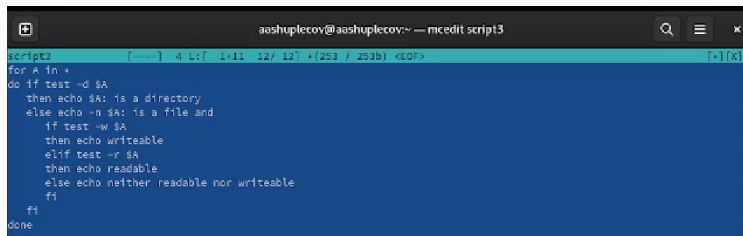
Рис. 4: текст скрипта, обрабатывающего любое произвольное число аргументов

Убедимся, что скрипт, обрабатывающий любое произвольное число аргументов, работает.

```
[aashuplecov@aashuplecov ~]$ mcedit script2  
  
[aashuplecov@aashuplecov ~]$ chmod +x script2  
[aashuplecov@aashuplecov ~]$ ./script2 1 2 3 4 5 6 7 8 9 10 11 12  
1: 1  
2: 2  
3: 3  
4: 4  
5: 5  
6: 6  
7: 7  
8: 8  
9: 9  
10: 10  
11: 11  
12: 12
```

Рис. 5: проверка скрипта, обрабатывающего любое произвольное число аргументов

Напишем командный файл — аналог команды `ls` (без использования самой этой команды и команды `dir`).



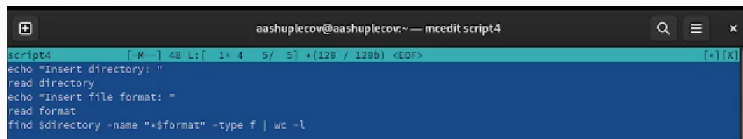
```
script3 [-----] 4 L: [ 1*11 12/ 12] *(253 / 253b) <EOF> [x] [X]
for A in *
do if test -d $A
then echo $A: is a directory
else echo -n $A: is a file and
if test -w $A
then echo writeable
elif test -r $A
then echo readable
else echo neither: readable nor writeable
fi
fi
done
```

Рис. 6: текст аналога команды `ls`

Убедимся, что командный файл - аналог команды ls, работает.

```
[aashuplecov@aashuplecov ~]$ mcdit script3
[aashuplecov@aashuplecov ~]$ chmod +x script3
[aashuplecov@aashuplecov ~]$ ./script3
abc1: is a file andwriteable
australia: is a directory
backup: is a directory
bin: is a directory
conf.txt: is a file andwriteable
ed: is a file andwriteable
ed.pub: is a file andwriteable
feathers: is a file andwriteable
file.txt: is a file andwriteable
elab07018: is a file andwriteable
lab07.sh: is a file andwriteable
lab07.sh~: is a file andwriteable
may: is a file andwriteable
monthly: is a directory
my_os: is a file andreadable
otchet: is a directory
play: is a directory
presentation: is a directory
reports: is a directory
rsa: is a file andwriteable
rsa.pub: is a file andwriteable
script1: is a file andwriteable
script2: is a file andwriteable
script3: is a file andwriteable
ski.places: is a directory
work: is a directory
video: is a directory
Документы: is a directory
Загрузки: is a directory
Изображения: is a directory
Музыка: is a directory
Общедоступные: is a directory
./script3: строка 2: test: Рабочий: ожидается бинарный оператор
Рабочий стол: is a file and./script3: строка 5: test: Рабочий: ожидается бинарный оператор
./script3: строка 7: test: Рабочий: ожидается бинарный оператор
neither readable nor writeable
[...]
```

Напишем 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.



```
script4 [M--] 48 L: [ 1* 4 5/ 5] *(12B / 12Bb) <EOF> [*] [X]
echo "insert directory: "
read directory
echo "insert file format: "
read format
find $directory -name "$format" -type f | wc -l
```

Рис. 8: текст командного файла, вычисляющего кол-во файлов в директории

Убедимся, что 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, работает.

```
[aashuplecov@aashuplecov ~]$ mcedit script4  
  
[aashuplecov@aashuplecov ~]$ chmod +x script4  
[aashuplecov@aashuplecov ~]$ ./script4  
Insert directory:  
/home/aashuplecov  
Insert file format:  
.md  
78
```

Рис. 9: проверка командного файла, вычисляющего кол-во файлов в директории

Я изучил основы программирования в оболочке ОС UNIX/Linux, научился писать небольшие командные файлы.

Кулябов Д.С. “Материалы к лабораторным работам”