

Лабораторная работа №10

Программирование в командном процессоре ОС UNIX. Командные файлы

Захаренко А.В.

17.03.2023

Российский университет дружбы народов, Москва, Россия

Информация

- Захаренко Анастасия Викторовна
- студентка первого курса бакалавриата
- направление бизнес-информатика, факультет физико-математических и естественных наук
- Российский университет дружбы народов

Элементы презентации

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 в другую директорию backup в вашем домашнем каталоге.
2. Написать пример командного файла, обрабатывающего любое произвольное число аргументов командной строки, в том числе превышающее десять.
3. Написать командный файл — аналог команды ls
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.

Командный процессор (командная оболочка, интерпретатор команд shell) — это программа, позволяющая пользователю взаимодействовать с операционной системой компьютера. В операционных системах типа UNIX/Linux наиболее часто используются следующие реализации командных оболочек: – оболочка Борна (Bourne shell или sh) — стандартная командная оболочка UNIX/Linux, содержащая базовый, но при этом полный набор функций; – C-оболочка (или csh) — надстройка на оболочке Борна, использующая C-подобный синтаксис команд с возможностью сохранения истории выполнения команд; – оболочка Корна (или ksh) — напоминает оболочку C, но операторы управления программой совместимы с операторами оболочки Борна; – BASH — сокращение от Bourne Again Shell (опять оболочка Борна), в основе своей совмещает свойства оболочек C и Корна (разработка компании Free Software Foundation).

POSIX (Portable Operating System Interface for Computer Environments) — набор стандартов описания интерфейсов взаимодействия операционной системы и прикладных программ. Стандарты POSIX разработаны комитетом IEEE (Institute of Electrical and Electronics Engineers) для обеспечения совместимости различных UNIX/Linux-подобных операционных систем и переносимости прикладных программ на уровне исходного кода. POSIX-совместимые оболочки разработаны на базе оболочки Корна.

1. Напишем скрипт, который при запуске будет делать резервную копию самого себя в другую директорию backup в вашем домашнем каталоге.

NAME

tar - an archiving utility

SYNOPSIS**Traditional usage**

tar {A|c|d|r|t|u|x}[GnSkUW0mpsMBiajJzZhPlRvwo] [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...]

```
[avzakharenko@fedora ~]$ touch script.sh  
[avzakharenko@fedora ~]$ chmod +x script.sh
```

Рис. 2: script

```
#!/bin/bash  
backupname="ScriptBack.sh"  
cp"$0" "$backup_name"  
tar -cf laba.tar $backup_name|
```

Рис. 3: script2

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

```
#!/bin/bash  
echo "Введите последовательность чисел"  
head -1|
```

Рис. 4: file

```
[avzakharenko@fedora ~]$ touch file.sh
[avzakharenko@fedora ~]$ chmod +x file.sh
[avzakharenko@fedora ~]$ ./file.sh
Введите последовательность чисел
11 12 13 14 15 16 17
11 12 13 14 15 16 17
[avzakharenko@fedora ~]$ ./file.sh
Введите последовательность чисел
6 7 8 9
6 7 8 9
```

Рис. 5: file02

3. Напишем командный файл — аналог команды ls


```
#!/bin/bash
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
```

```
[avzakharenko@fedora ~]$ touch file2.sh
[avzakharenko@fedora ~]$ chmod +x file2.sh
[avzakharenko@fedora ~]$ ./file2.sh
australia: is a directory
conf.txt: is a file andwriteable
feathers: is a file andwriteable
file2.sh: is a file andwriteable
file.sh: is a file andwriteable
file.txt: 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
```

4. 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.

```
#!/bin/bash
direct=''
form=''
echo 'write format'
read form
echo 'write directory'
read direct
find "$direct" -name "$form" -type f | wc -l
ls
```

```
file3.sh      may      script.sh      Видео      Общедоступные
[avzakharenko@fedora ~]$ ./file3.sh
write format
sh
write directory
play
3
australia    file.sh      monthly     ski.plases   Документы    'Рабочий стол'
conf.txt     file.txt     my_os       text.txt     Загрузки     Шаблоны
feathers     lab07.sh     play        wew          Изображения
file2.sh     lab07.sh~   reports     work         Музыка
file3.sh     may         script.sh   Видео        Общедоступные
```

Рис. 9: file3.02

Я изучила основы программирования в оболочке ОС UNIX/Linux. Научилась писать небольшие командные файлы