

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Доленко Дарья Васильевна НБИбд-01-21¹

28 мая, 2022, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

Изучить основы программирования в оболочке ОС UNIX.
Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Процесс выполнения лабораторной работы

`(../report/image/1.jpg){#fig:001 width=70%}`

Используя команды `getopts` `grep`, написала командный файл, который анализирует командную строку с ключами: `-i`inputfile; `-o`outputfile; `-p`шаблон; `-C`; `-n`, затем ищет в указанном файле нужные строки, определяемые ключом `-p`. (рис. 1 2 3)

```
[dvdolenko@fedora ~]$ bash lab11.sh -ilab11.txt -olab11-1.txt -pice
[dvdolenko@fedora ~]$ cat ~/lab11.txt
Дале-далеко, в той стране, где всходит солнце
Ранним утром милый больше не проснётся
Льётся алая река, лиса смеётся
С восточным ветром пусть сказание несётся
В забытом ныне городе жила девица
Великим самураям от любви не спится
Лица жадные горят. Как не влюбиться?
Но в хитрых девичьих глазах кошмар таится
Не беги за нею, глупый
По её следам идут лишь трупы
Зубы крепче, чем металл
Ты попал!
[dvdolenko@fedora ~]$ ./lab11.sh -i ~/lab11.txt -o ~/lab11-1.txt -p C -C -n
[dvdolenko@fedora ~]$ cat ~/lab11-1.txt
Дале-далеко, в той стране, где всходит солнце
Ранним утром милый больше не проснётся
Льётся алая река, лиса смеётся
С восточным ветром пусть сказание несётся
Великим самураям от любви не спится
Лица жадные горят. Как не влюбиться?
Но в хитрых девичьих глазах кошмар таится
По её следам идут лишь трупы
[dvdolenko@fedora ~]$ ./lab11.sh -i ~/lab11.txt -o ~/lab11-1.txt -p C -n
[dvdolenko@fedora ~]$ cat ~/lab11-1.txt
4:С восточным ветром пусть сказание несётся
[dvdolenko@fedora ~]$ ./lab11.sh -i ~/lab11.txt -C -n
./lab11.sh: строка 13: {/: --0: синтаксическая ошибка: ожидается operand (чужеродный
```

```
1 Дале-далеко, в той стране, где всходит солнце
2 Ранним утром милый больше не проснётся
3 Льётся алая река, лиса смеётся
4 С восточным ветром пусть сказание несётся
5 В забытом ныне городе жила девица
6 Великим самураям от любви не спится
7 Лица жадные горят. Как не влюбиться?
8 Но в хитрых девичьих глазах кошмар таится
9 Не беги за нею, глупый
10 По её следам идут лишь трупы
11 Зубы крепче, чем металл
12 Ты попал!
```

Figure 2: Содержание текстового файла

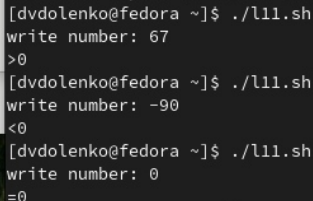
```

1 #!/bin/bash
2 inflag=0; oflag=0; pflag=0; cflag=0; nflag=0;
3 while getopts i:op:C:n optletter
4 do case $optletter in
5     i) iflag=1; ival=$OPTARG;;
6     o) oflag=1; oval=$OPTARG;;
7     p) pflag=1; pval=$OPTARG;;
8     C) cflag=1;;
9     n) nflag=1;;
10    *) echo illegal option $optletter
11    esac
12 done
13 if (($pflag==0))
14 then echo "not found"
15 else
16     if (($iflag==0))
17     then echo "file not found"
18     else
19         if (($oflag==0))
20         then if ((cflag==0))
21             then if ((nflag==0))
22                 then grep $pval $ival
23                 else grep -n $pval $ival
24                 fi
25             else if ((nflag==0))
26                 then grep -i $pval $ival
27                 else grep -i -n $pval $ival
28                 fi
29             fi
30         else if ((cflag==0))
31             then if ((nflag==0))
32                 then grep $pval $ival > $oval
33                 else grep -n $pval $ival > $oval
34                 fi
35             else if ((nflag==0))
36                 then grep -i $pval $ival > $oval
37                 else grep -i -n $pval $ival > $oval
38                 fi
39             fi
40         fi
41     fi
42 fi

```

Figure 3: Текст программы

Написала на языке Си программу, которая вводит число и определяет, является ли оно больше нуля, меньше нуля или равно нулю. Затем программа завершается с помощью функции `exit(n)`, передавая информацию в о коде завершения в оболочку. Командный файл должен вызывать эту программу и, проанализировав с помощью команды `$?`, выдать сообщение о том, какое число было введено. (рис. 4 5 6)



```
[dvdolenko@fedora ~]$ ./l11.sh
write number: 67
>0
[dvdolenko@fedora ~]$ ./l11.sh
write number: -90
<0
[dvdolenko@fedora ~]$ ./l11.sh
write number: 0
=0
```

Figure 4: Пример работы программы

```
1 #!/bin/bash
2
3 gcc l11.c -o l11
4 ./l11
5 code=$?
6 case $code in
7     0) echo "<0";;
8     1) echo ">0";;
9     2) echo "=0";;
10 esac
```

Figure 5: Файл sh

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main() {
5     printf("write number: ");
6     int n;
7     scanf("%d", &n);
8     if (n<0) exit(0);
9     if (n>0) exit(1);
10    if (n==0) exit(2);
11    return 0;
12 }
```

Figure 6: Файл С

Написала командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до X (например 1.tmp, 2.tmp, 3.tmp, 4.tmp и т.д.). Число файлов, которые необходимо создать, передаётся в аргументы командной строки. Этот же командный файл должен уметь удалять все созданные им файлы (если они существуют). (рис. 7 8 9)

```
[dvdolenko@fedora ~]$ touch llll.sh
[dvdolenko@fedora ~]$ chmod +x *.sh
[dvdolenko@fedora ~]$ ls
abc1.cpp      lll      may      ski.places  Музыка
australia    lll.c     monthly  snap        Общедоступные
backup       lll1.sh   my_os    text.txt    'Рабочий стол'
backup.sh     '#lab07.sh#'  play     tutorial     c.sh
b.sh         lab07.sh  reports  work        c.sh~
b.sh~       lab11-1.txt  script.sh  Видео       Шаблоны
conf.txt     lab11.sh  script.sh~  Документы
feathers     lab11.txt  scriptttt.sh  Загрузки
fille.txt    lll1.sh   scriptttt.sh~  Изображения
```

Figure 7: Пример работы программы (состояние до применения)

```
[dvdolenko@fedora ~]$ ./l111.sh -c VD#.txt 8
[dvdolenko@fedora ~]$ la
bash: la: command not found...
[dvdolenko@fedora ~]$ ls
```

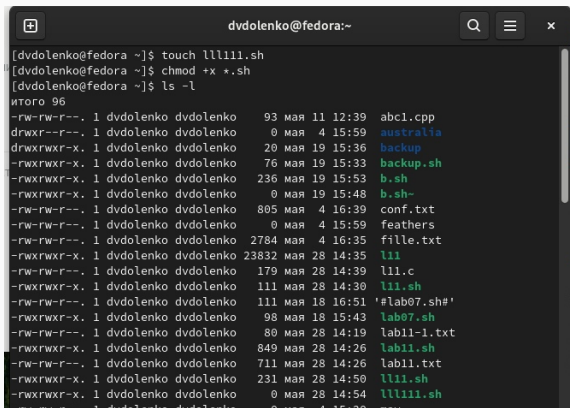
abc1.cpp	l11.c	my_os	tutorial	Видео
australia	l11.sh	play	VD1.txt	Документы
backup	'#lab07.sh#'	reports	VD2.txt	Загрузки
backup.sh	lab07.sh	script.sh	VD3.txt	Изображения
b.sh	lab11-1.txt	script.sh~	VD4.txt	Музыка
b.sh~	lab11.sh	scriptttt.sh	VD5.txt	Общедоступные
conf.txt	lab11.txt	scriptttt.sh~	VD6.txt	'Рабочий стол'
feathers	l111.sh	ski.plases	VD7.txt	c.sh
fil1e.txt	may	snap	VD8.txt	c.sh~
l11	monthly	text.txt	work	Шаблоны

Figure 8: Пример работы программы (после применения)

```
1 #!/bin/bash
2
3 opt=$1;
4 form=$2;
5 num=$3;
6 function Files() {
7     for ((i=1; i<=$num; i++)) do
8         file=$(echo $form | tr '#' "$i")
9         if [ $opt == "-r" ]
10         then
11             rm -f $file
12         elif [ $opt == "-c" ]
13         then
14             touch $file
15         fi
16     done
17 }
18 Files
```

Figure 9: Текст программы

Написала командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировала его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`). (рис. 10 11 12 13)



```
[dvdolenko@fedora ~]$ touch llllll.sh
[dvdolenko@fedora ~]$ chmod +x *.sh
[dvdolenko@fedora ~]$ ls -l
итого 96
-rw-rw-r--. 1 dvdolenko dvdolenko   93 мая 11 12:39 abc1.cpp
drwxr--r--. 1 dvdolenko dvdolenko    0 мая  4 15:59 australia
drwxrwxr-x. 1 dvdolenko dvdolenko   20 мая 19 15:36 backup
-rwxrwxr-x. 1 dvdolenko dvdolenko   76 мая 19 15:33 backup.sh
-rwxrwxr-x. 1 dvdolenko dvdolenko  236 мая 19 15:53 b.sh
-rwxrwxr-x. 1 dvdolenko dvdolenko    0 мая 19 15:48 b.sh-
-rw-rw-r--. 1 dvdolenko dvdolenko  805 мая  4 16:39 conf.txt
-rw-rw-r--. 1 dvdolenko dvdolenko    0 мая  4 15:59 feathers
-rw-rw-r--. 1 dvdolenko dvdolenko 2784 мая  4 16:35 fille.txt
-rwxrwxr-x. 1 dvdolenko dvdolenko 23832 мая 28 14:35 l11
-rw-rw-r--. 1 dvdolenko dvdolenko   179 мая 28 14:39 l11.c
-rwxrwxr-x. 1 dvdolenko dvdolenko   111 мая 28 14:30 l11.sh
-rw-rw-r--. 1 dvdolenko dvdolenko   111 мая 18 16:51 '#lab07.sh#'
-rwxrwxr-x. 1 dvdolenko dvdolenko    98 мая 18 15:43 lab07.sh
-rw-rw-r--. 1 dvdolenko dvdolenko    80 мая 28 14:19 lab11-1.txt
-rwxrwxr-x. 1 dvdolenko dvdolenko   849 мая 28 14:26 lab11.sh
-rw-rw-r--. 1 dvdolenko dvdolenko   711 мая 28 14:26 lab11.txt
-rwxrwxr-x. 1 dvdolenko dvdolenko   231 мая 28 14:50 llllll.sh
-rwxrwxr-x. 1 dvdolenko dvdolenko    0 мая 28 14:54 llllll.sh
```

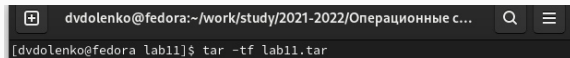
Figure 10: До применения программы

```
[dvdo@lenko@fedora ~]$ sudo ~/lll111.sh
Изображения/
Изображения/Снимок экрана от 2022-04-27 10-01-33.png
Изображения/Снимок экрана от 2022-04-27 10-01-41.png
Изображения/Снимок экрана от 2022-05-11 08-56-56.png
Изображения/Снимок экрана от 2022-05-11 09-44-51.png
Изображения/Снимок экрана от 2022-05-11 10-16-46.png
Изображения/Снимок экрана от 2022-05-11 10-18-56.png
Изображения/Снимок экрана от 2022-05-11 10-22-03.png
Изображения/Снимок экрана от 2022-05-11 10-23-10.png
Изображения/Снимок экрана от 2022-05-11 10-25-25.png
Изображения/Снимок экрана от 2022-05-11 11-00-06.png
Изображения/Снимок экрана от 2022-05-11 11-08-37.png
Изображения/Снимок экрана от 2022-05-11 12-40-01.png
Изображения/Снимок экрана от 2022-05-11 15-33-42.png
Изображения/Снимок экрана от 2022-05-28 14-22-34.png
Изображения/Снимок экрана от 2022-05-28 14-36-21.png
Изображения/Снимок экрана от 2022-05-28 14-53-20.png
.vboxclient-clipboard.pid
.vboxclient-seamless.pid
.vboxclient-draganddrop.pid
.bash_history
lab11-1.txt
lab11.txt
lab11.sh
ll1.sh
ll1
ll1.c
ll11.sh
VD1.txt
VD2.txt
VD3.txt
VD4.txt
VD5.txt
VD6.txt
VD7.txt
VD8.txt
lll111.sh
[dvdo@lenko@fedora ~]$
```

Figure 11: После применения программы


```
1 #!/bin/bash
2
3 files=$(find ./ -maxdepth 1 -mtime -7)
4 listing=""
5 for file in "$files" ; do
6     file=$(echo "$file" | cut -c 3-)
7     listing="$listing $file"
8 done
9 dir=$(basename $(pwd))
10 tar -cvf $dir.tar $listing
```

Figure 12: Текст программы

A terminal window with a dark background. The title bar shows a plus icon, the text 'dvdolenko@fedora:~/work/study/2021-2022/Операционные с...', a magnifying glass icon, and a hamburger menu icon. The terminal text shows the prompt '[dvdolenko@fedora lab11]\$' followed by the command 'tar -tf lab11.tar'.

```
[dvdolenko@fedora lab11]$ tar -tf lab11.tar
```

Figure 13: Команды архивации

Выводы по проделанной работе

В ходе данной лабораторной работы я изучила основы программирования в оболочке ОС UNIX, научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.