

Презентация по лабораторной работе №11

Программирование в командном процессоре ОС UNIX. Ветвления и циклы

Сагдеров Камал

19.04.2023

Российский университет дружбы народов, Москва, Россия

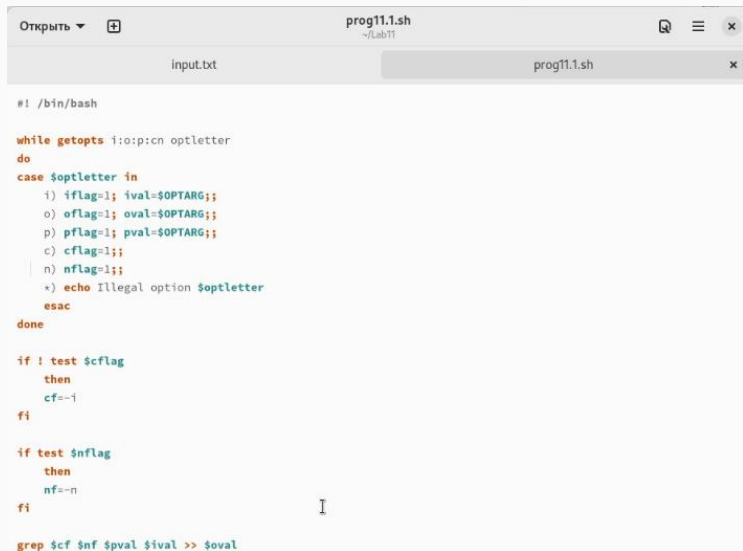
Изучить основы программирования в оболочке ОС UNIX. Научится писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: `-i`inputfile — прочитать данные из указанного файла; `-o`outputfile — вывести данные в указанный файл; `-r`шаблон — указать шаблон для поиска; `-C` — различать большие и малые буквы; `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-r`.
2. Написать 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.

3. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до `N` (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).
4. Написать командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`).

1. Используя команды `getopts` `grep`, написать командный файл, который анализирует командную строку с ключами: – `-iinputfile` — прочитать данные из указанного файла; – `-ooutputfile` — вывести данные в указанный файл; – `-rшаблон` — указать шаблон для поиска; – `-C` — различать большие и малые буквы; – `-n` — выдавать номера строк. а затем ищет в указанном файле нужные строки, определяемые ключом `-r`.

Процесс выполнения



```
#!/bin/bash

while getopts i:o:p:cn optletter
do
case $optletter in
i) iflag=1; ival=$OPTARG;;
o) oflag=1; oval=$OPTARG;;
p) pflag=1; pval=$OPTARG;;
c) cflag=1;;
n) nflag=1;;
*) echo Illegal option $optletter
esac
done

if ! test $cflag
then
cf=-i
fi

if test $nflag
then
nf=-n
fi

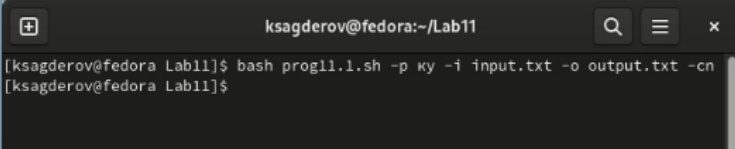
grep $cf $nf $pval $ival >> $oval
```



Рис. 2: Результат



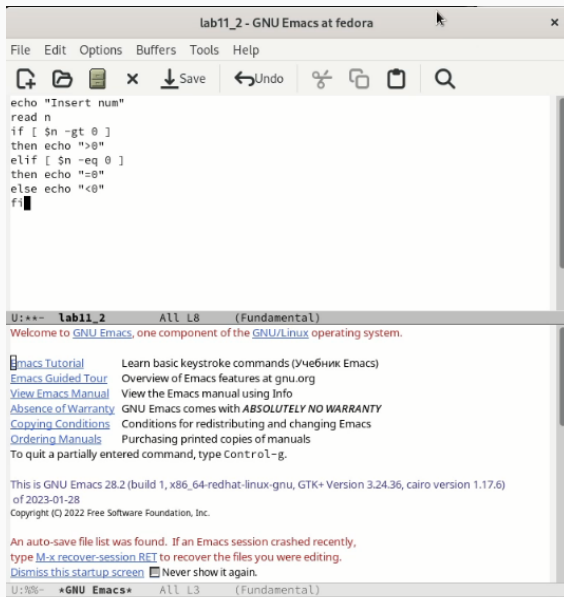
Рис. 3: Результат

A terminal window with a dark background. The title bar at the top shows a plus icon on the left, the text 'ksagderov@fedora:~/Lab11' in the center, and search, menu, and close icons on the right. The terminal content shows a prompt '[ksagderov@fedora Lab11]\$' followed by the command 'bash prog11.1.sh -p ку -i input.txt -o output.txt -cn'. The next line shows the prompt again, indicating the command has been executed.

```
ksagderov@fedora:~/Lab11
[ksagderov@fedora Lab11]$ bash prog11.1.sh -p ку -i input.txt -o output.txt -cn
[ksagderov@fedora Lab11]$
```

Рис. 4: Вызов программы в терминале

2. Написать 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



```
echo "Insert num"
read n
if [ $n -gt 0 ]
then echo ">0"
elif [ $n -eq 0 ]
then echo "=0"
else echo "<0"
fi
```

U:***- lab11_2 All L8 (Fundamental)

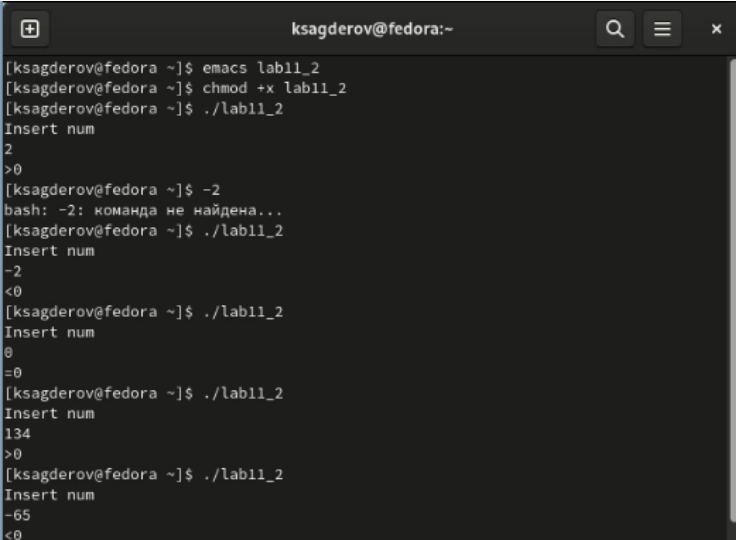
Welcome to [GNU Emacs](#), one component of the [GNU/Linux](#) operating system.

[Emacs Tutorial](#) Learn basic keystroke commands (Учебник Emacs)
[Emacs Guided Tour](#) Overview of Emacs features at gnu.org
[View Emacs Manual](#) View the Emacs manual using Info
[Absence of Warranty](#) GNU Emacs comes with **ABSOLUTELY NO WARRANTY**
[Copying Conditions](#) Conditions for redistributing and changing Emacs
[Ordering Manuals](#) Purchasing printed copies of manuals
To quit a partially entered command, type `Control-g`.

This is GNU Emacs 28.2 (build 1, x86_64-redhat-linux-gnu, GTK+ Version 3.24.36, cairo version 1.17.6)
of 2023-01-28
Copyright (C) 2022 Free Software Foundation, Inc.

An auto-save file list was found. If an Emacs session crashed recently,
type `M-x recover-session RET` to recover the files you were editing.
[Dismiss this startup screen](#) ☐ Never show it again.

U:***- *GNU Emacs* All L3 (Fundamental)



```
ksagderov@fedora:~  
[ksagderov@fedora ~]$ emacs lab11_2  
[ksagderov@fedora ~]$ chmod +x lab11_2  
[ksagderov@fedora ~]$ ./lab11_2  
Insert num  
2  
>0  
[ksagderov@fedora ~]$ -2  
bash: -2: команда не найдена...  
[ksagderov@fedora ~]$ ./lab11_2  
Insert num  
-2  
<0  
[ksagderov@fedora ~]$ ./lab11_2  
Insert num  
0  
=0  
[ksagderov@fedora ~]$ ./lab11_2  
Insert num  
134  
>0  
[ksagderov@fedora ~]$ ./lab11_2  
Insert num  
-65  
<0
```

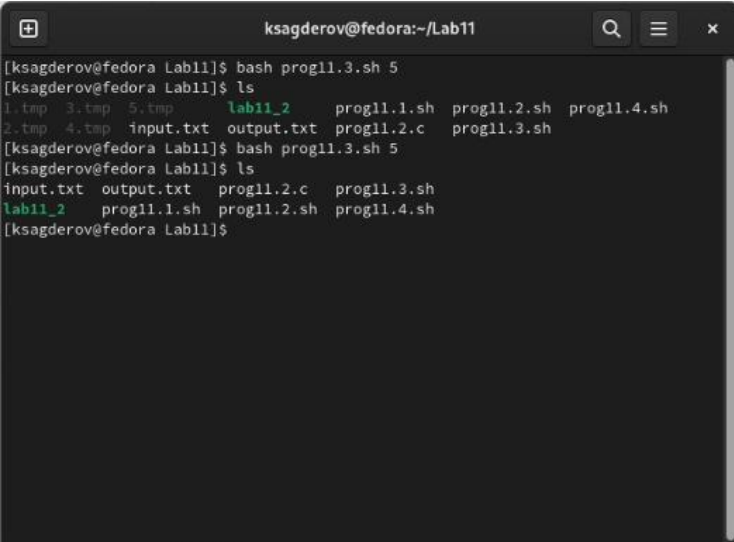
Рис. 6: Результат

3. Написать командный файл, создающий указанное число файлов, пронумерованных последовательно от 1 до `N` (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)



```
Открыть ▾ (+) prog11.3.sh ~/Lab11
#!/bin/bash
for ((i=1; i<=*; i++))
do
if test -f "$i".tmp
then rm "$i".tmp
else touch "$i.tmp"
fi
done
```

Рис. 7: Третья программа



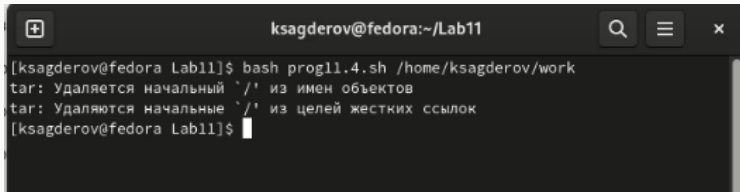
```
ksagderov@fedora:~/Lab11
[ksagderov@fedora Lab11]$ bash prog11.3.sh 5
[ksagderov@fedora Lab11]$ ls
1.tmp  3.tmp  5.tmp  lab11_2  prog11.1.sh  prog11.2.sh  prog11.4.sh
2.tmp  4.tmp  input.txt  output.txt  prog11.2.c  prog11.3.sh
[ksagderov@fedora Lab11]$ bash prog11.3.sh 5
[ksagderov@fedora Lab11]$ ls
input.txt  output.txt  prog11.2.c  prog11.3.sh
lab11_2    prog11.1.sh  prog11.2.sh  prog11.4.sh
[ksagderov@fedora Lab11]$
```

4. Написать командный файл, который с помощью команды `tar` запаковывает в архив все файлы в указанной директории. Модифицировать его так, чтобы запаковывались только те файлы, которые были изменены менее недели тому назад (использовать команду `find`)



```
Открыть ▾ + prog11.4.sh  
~/.Lab11  
#!/bin/bash  
  
find $* -mtime -7 -mtime +0 -type f > FILES.txt  
tar -cf archive.tar -T FILES.txt
```

Рис. 9: Четвертая программа



A terminal window titled "ksagderov@fedora:~/Lab11" with search, menu, and close icons. The terminal shows the execution of a script "prog11.4.sh" in the directory "/home/ksagderov/work". The script outputs two lines of Russian text: "tar: Удаляется начальный '/' из имен объектов" and "tar: Удаляются начальные '/' из целей жестких ссылок". The prompt returns to the user.

```
[ksagderov@fedora Lab11]$ bash prog11.4.sh /home/ksagderov/work
tar: Удаляется начальный '/' из имен объектов
tar: Удаляются начальные '/' из целей жестких ссылок
[ksagderov@fedora Lab11]$
```

Рис. 10: Результат

Процесс выполнения

```
Открыть ▾ + FILES.txt ~/Lab11
prog11.4.sh FILES.txt x

/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/refs/heads/master
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/refs/remotes/origin/master
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/8c/
98eccc828a7bd78945d69498adae9c65b95e
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/7b/
7706ebf3c559018a34873f15528ca2bd62add
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/8f/
feb50f0d91c3773c99f1739023d8efeb34f3
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
72/4e2c205c4d93a492d36570hfee29ca6d3e3356
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/fc/
9d56e91ac76884927053a2e4356b1ec363dbcc5
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/5a/
713d9ce0ecd0648e769ebd7b3320603878e72b
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/4c/
35c4ea729d0aaf5c198f42bd1a47b797179e
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/b9/
b999ed344a96ab8c970c2251d901da46f8d6c9
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
b9/6c756782d38bbf5312123833e7a27886a94cdh
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
92/7cdf56c64d2c4eb095131124024274a190f1d5
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/50/
f48e5c5e3a6929880884af0f912b341daa180a
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
84/9083e626c7d0dd8d01d6188954d2c35963d8c3
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/cf/
8be87474e116180d45ba3183543b232558d6b2
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
22/44a788592a792c64f0973468be31dc9bb1eb24
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/ae/
8af4b6a0661201f591cb36bdf03c226199e183
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/de/
ad5e4c87e9409061c9c37c245706fcc67ca8
/home/ksaaderov/work/study/2022-2023/Операционные системы/os-intro/.git/objects/
22/06723232a34f40441370e72402bb4020f72d
```

В процессе выполнения данной лабораторной работы я изучил основы программирования в оболочке ОС UNIX. Научился писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

Спасибо за внимание!
