

# **ЛАБОРАТОРНАЯ РАБОТА №13**

Операционные системы

Богданюк А.В., НКАбд-01-23

Российский университет дружбы народов, Москва, Россия

09 марта 2024

## **ЦЕЛЬ РАБОТЫ**

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

## ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Для начала создаю файл p1.sh и делаю его исполняемым (рис. 1).

```
avbogdanyuk@Bogdanyuk:~/work/lab13$ touch p1.sh  
avbogdanyuk@Bogdanyuk:~/work/lab13$ chmod +x p1.sh  
avbogdanyuk@Bogdanyuk:~/work/lab13$ _
```

Создаю p1.sh

## **ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ**

Затем пишу в нем программный файл, который анализирует командную строку с ключами: – -iinputfile — прочитать данные из указанного файла; – -ooutputfile — вывести данные в указанный файл; – -ршаблон — указать шаблон для поиска; – -С — различать большие и малые буквы; – -n — выдавать номера строк, а затем ищет в указанном файле нужные строки, определяемые ключом -р(рис. 2).

```
avbogdanyuk@Bogdanyuk: ~/work/lab13
GNU nano 6.2 p1.sh *
while getopts i:o:p:c:n 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
```

Программный файл

# ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Теперь запускаю `bash p1.sh`, вывод корректен (рис. 3).

```
avbogdanyuk@Bogdanyuk:~/work/lab13$ bash p1.sh -p озёр -i input.txt -o output.txt -c -n
avbogdanyuk@Bogdanyuk:~/work/lab13$ cat output.txt
Вели́кие озёра[2], также Вели́кие америка́нские озёра[3] или Вели́кие североамерика́нские озёра[4] (англ. Great Lakes, фр. Grands Lacs), – система пресноводных озёр в Северной Америке, на территории США и Канады.
Включает ряд крупных и средних водоёмов, соединённых реками и проливами, которые образуют самую большую по площади группу пресноводных озёр на Земле (общая площадь более 244 тыс. км², общий объём 22,7 тыс. км³), содержащую 21 % поверхностных пресных вод в мире и 84 % – в Северной Америке. К собственно Великим озёрам относят пять крупнейших: Верхнее, Мичиган, Гурон, Эри и Онтарио.
Озеро Мичиган находится целиком в США, по остальным озёрам и соединяющим их коротким рекам проходит граница между США (% акватории озёр) и Канадой (% акватории). С системой Великих озёр связаны несколько средних озёр (в том числе Нипигон, Ниписсинг, Симко, Уиннебейго) и множество более мелких.
1:Вели́кие озёра[2], также Вели́кие америка́нские озёра[3] или Вели́кие североамерика́нские озёра[4] (англ. Great Lakes, фр. Grands Lacs), – система пресноводных озёр в Северной Америке, на территории США и Канады.
2:Включает ряд крупных и средних водоёмов, соединённых реками и проливами, которые образуют самую большую по площади группу пресноводных озёр на Земле (общая площадь более 244 тыс. км², общий объём 22,7 тыс. км³), содержащую 21 % поверхностных пресных вод в мире и 84 % – в Северной Америке. К собственно Великим озёрам относят пять крупнейших: Верхнее, Мичиган, Гурон, Эри и Онтарио.
4:Озеро Мичиган находится целиком в США, по остальным озёрам и соединяющим их коротким рекам проходит граница между США (% акватории озёр) и Канадой (% акватории). С системой Великих озёр связаны несколько средних озёр (в том числе Нипигон, Ниписсинг, Симко, Уиннебейго) и множество более мелких.
```

Вывод `p1.sh`

## ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

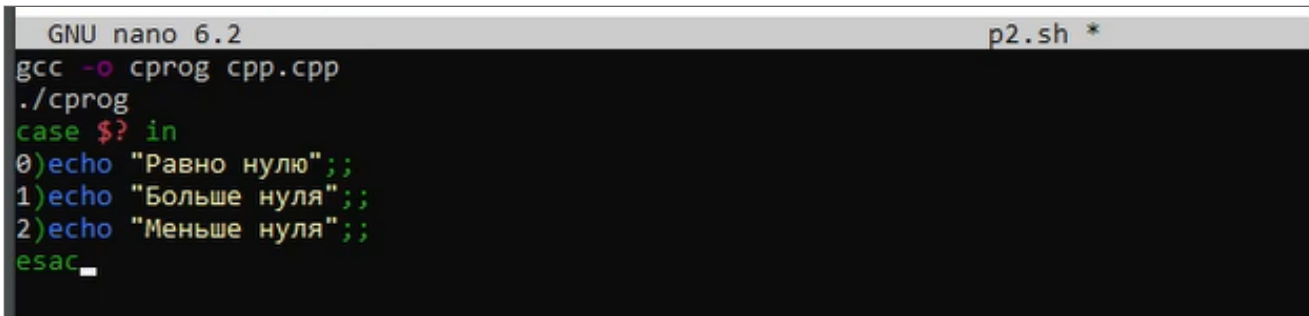
Создаю файл p2.sh и делаю его исполняемым (рис. 4).

```
avbogdanyuk@Bogdanyuk:~/work/lab13$ touch p2.sh  
avbogdanyuk@Bogdanyuk:~/work/lab13$ chmod +x p2.sh  
avbogdanyuk@Bogdanyuk:~/work/lab13$ _
```

Создаю p2.sh

# ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Теперь пишу программный файл, который л должен вызывать эту программу и, проанализировав с помощью команды \$?, выдать сообщение о том, какое число было введено (рис. 5).



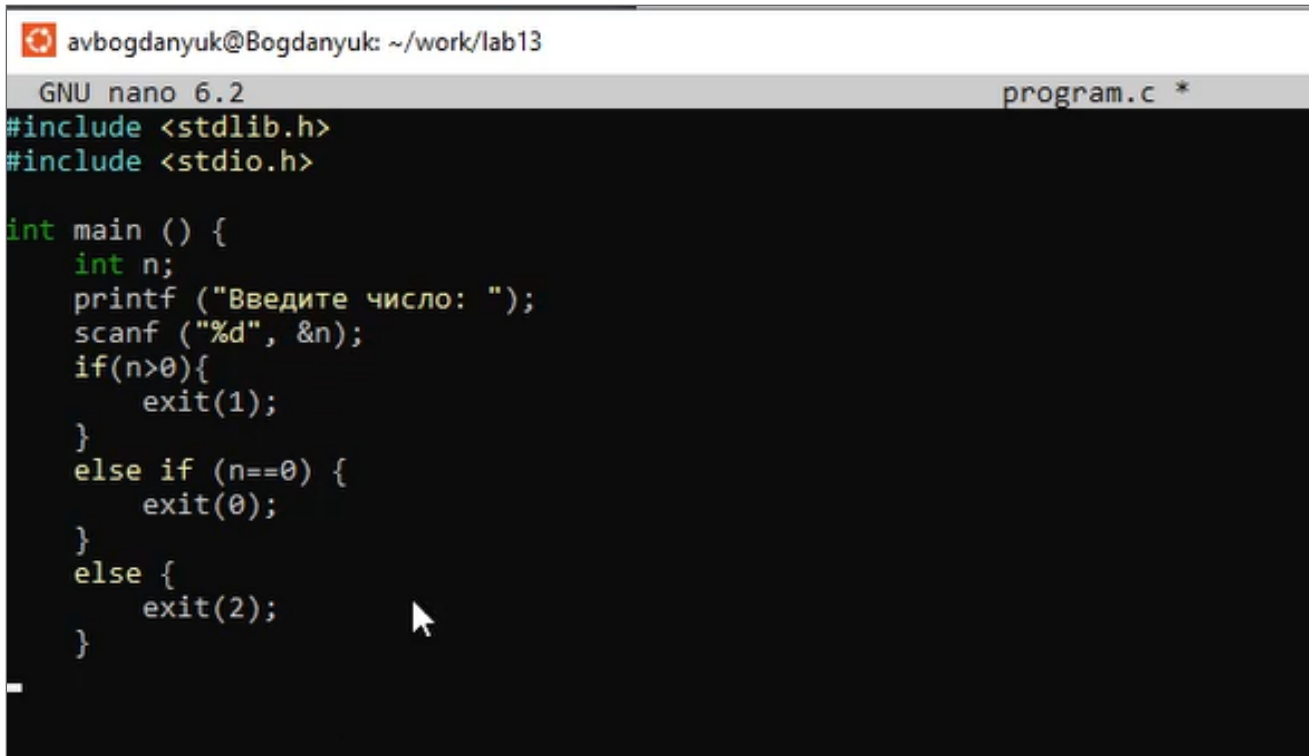
```
GNU nano 6.2 p2.sh *
gcc -o cprog cpp.cpp
./cprog
case $? in
0)echo "Равно нулю";;
1)echo "Больше нуля";;
2)echo "Меньше нуля";;
esac_
```

Программный файл



# ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Пишу 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. (рис. 6).

A screenshot of a terminal window. The title bar shows the user 'avbogdanyuk' at host 'Bogdanyuk' in the directory '~/work/lab13'. The terminal content shows the GNU nano 6.2 editor editing a file named 'program.c'. The code is a C program that prompts the user to enter a number, reads it, and then uses the 'exit' function to terminate with a status code based on whether the number is positive, zero, or negative.

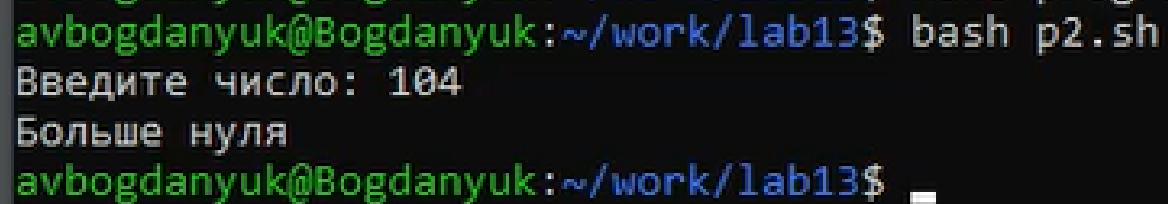
```
avbogdanyuk@Bogdanyuk: ~/work/lab13
GNU nano 6.2                                program.c *
#include <stdlib.h>
#include <stdio.h>

int main () {
    int n;
    printf ("Введите число: ");
    scanf ("%d", &n);
    if(n>0){
        exit(1);
    }
    else if (n==0) {
        exit(0);
    }
    else {
        exit(2);
    }
}
```

# Програма на C

## ВЫПОЛНЕНИЕ ЛАБОРАТОРНОЙ РАБОТЫ

Вывод bash p2.sh, вывод корректен (рис. 7).

A terminal window with a black background and green text. The prompt is 'avbogdanyuk@Bogdanyuk:~/work/lab13\$'. The user enters 'bash p2.sh'. The script outputs 'Введите число: 104' and 'Больше нуля'. The prompt returns to 'avbogdanyuk@Bogdanyuk:~/work/lab13\$' followed by a cursor.

```
avbogdanyuk@Bogdanyuk:~/work/lab13$ bash p2.sh
Введите число: 104
Больше нуля
avbogdanyuk@Bogdanyuk:~/work/lab13$ _
```

Вывод p2.sh

## **ВЫВОД**

В ходе выполнения лабораторной работы были изучены основы программирования в оболочке ОС UNIX. Научилась писать более сложные командные файлы с использованием логических управляющих конструкций и циклов.

# СПИСОК ЛИТЕРАТУРЫ

1. Dash P. Getting started with oracle vm virtualbox. Packt Publishing Ltd, 2013. 86 p.
2. Colvin H. Virtualbox: An ultimate guide book on virtualization with virtualbox. CreateSpace Independent Publishing Platform, 2015. 70 p.
3. van Vugt S. Red hat rhcsa/rhce 7 cert guide : Red hat enterprise linux 7 (ex200 and ex300). Pearson IT Certification, 2016. 1008 p.
4. Робачевский А., Немнюгин С., Стесик О. Операционная система unix. 2-е изд. Санкт-Петербург: БХВ-Петербург, 2010. 656 p.
5. Немец Э. et al. Unix и Linux: руководство системного администратора. 4-е изд. Вильямс, 2014. 1312 p.
6. Колисниченко Д.Н. Самоучитель системного администратора Linux. СПб.: БХВ-Петербург, 2011. 544 p.
7. Robbins A. Bash pocket reference. O'Reilly Media, 2016. 156 p.