

Липецкий государственный технический университет

Факультет автоматизации и информатики
Кафедра автоматизированных систем управления

Отчет по лабораторной работе № 5
по дисциплине «OS Linux»
на тему «Программирование на SHELL. Использование командных файлов»

Студент

Группа АС-18-1

Руководитель

К.Н.

учёная степень, учёное звание

подпись, дата

подпись, дата

Сухоруков К.О.

фамилия, инициалы

Кургасов В.В.

фамилия, инициалы

Липецк 2020 г.

СОДЕРЖАНИЕ

| | |
|-------------------------------|----|
| Цель работы | 2 |
| 1 Ход работы | 3 |
| Вывод | 29 |
| Контрольные вопросы | 30 |

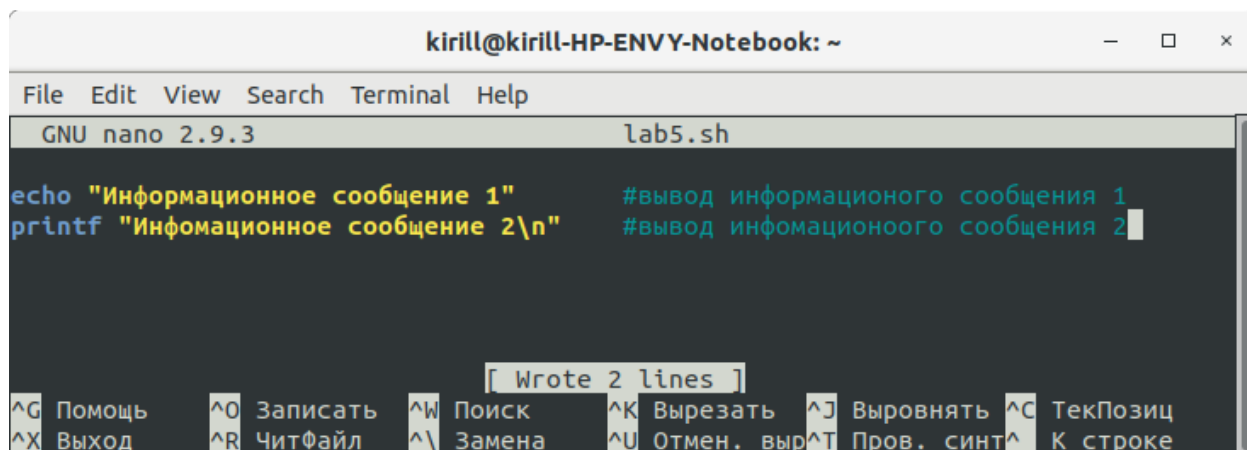
Цель работы

Изучение основных возможностей языка программирования Shell с целью автоматизации процесса администрирования системы за счет написания и использования командных файлов.

1 Ход работы

1. Используя команды ECHO, PRINTF вывести информационные сообщения на экран

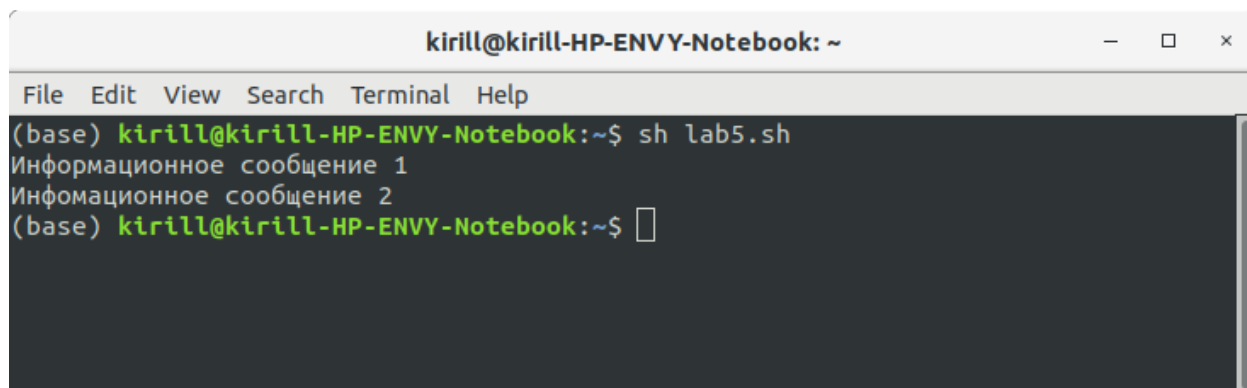
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.1.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh
echo "Информационное сообщение 1"      #вывод информационного сообщения 1
printf "Информационное сообщение 2\n"  #вывод инфомационного сообщения 2
[ Wrote 2 lines ]
^G Помощь      ^O Записать    ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход      ^R ЧитФайл    ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.1 – Текст скрипта для задания 1

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.2.

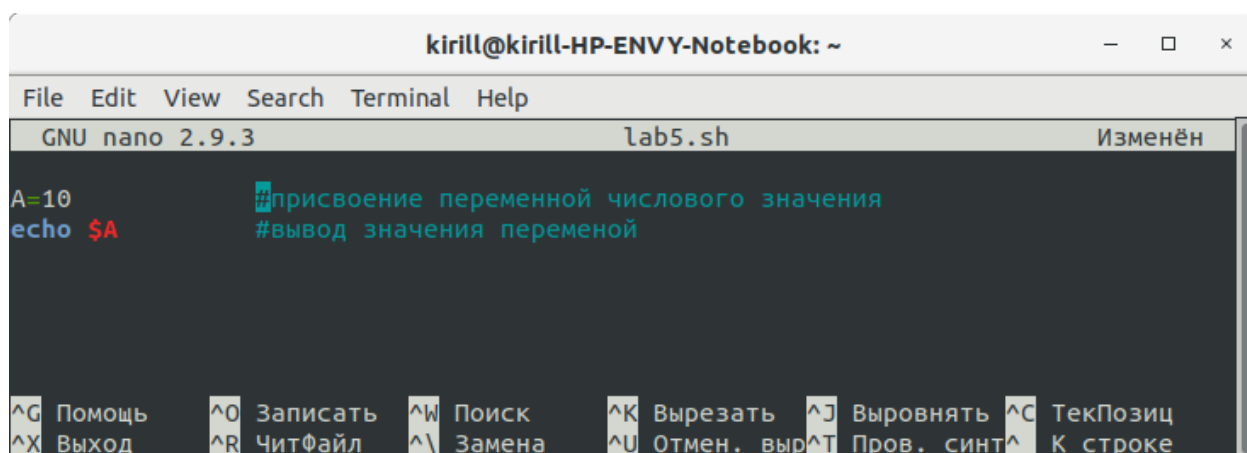


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Информационное сообщение 1
Информационное сообщение 2
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.2 – Результат выполнения скрипта для задания 1

2. Присвоить переменной А целочисленное значение. Просмотреть значение переменной А

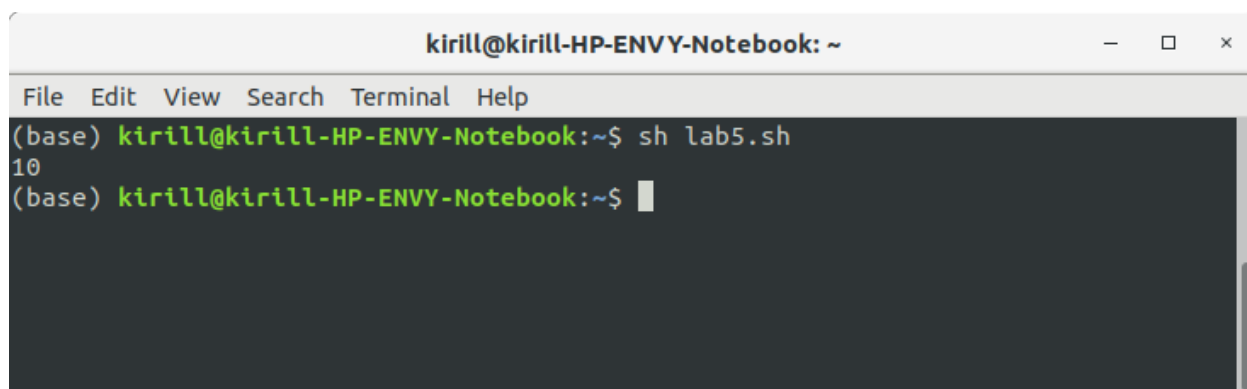
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.3.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh Изменён  
A=10      #присвоение переменной числового значения  
echo $A   #вывод значения переменных  
^G Помощь  ^O Записать  ^W Поиск     ^K Вырезать  ^J Выводить  ^C ТекПозиц  
^X Выход    ^R ЧитФайл  ^\ Замена    ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.3 – Текст скрипта для задания 2

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.4.

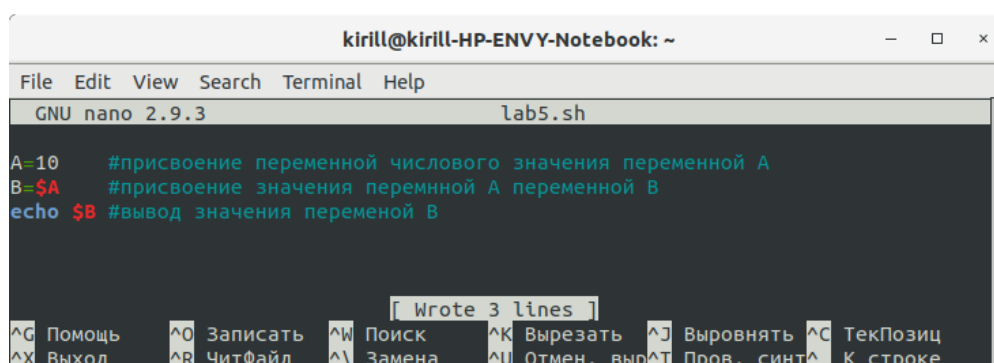


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
10  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.4 – Результат выполнения скрипта для задания 2

3. Присвоить переменной `B` значение переменной `A`. Просмотреть значение переменной `B`

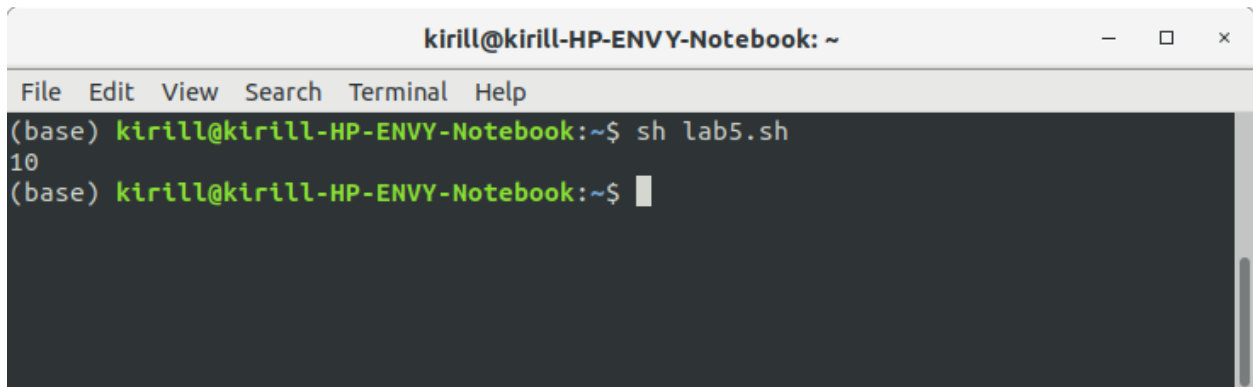
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.5.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh  
A=10      #присвоение переменной числового значения переменной A  
B=$A      #присвоение значения переменной A переменной B  
echo $B   #вывод значения переменной B  
[ Wrote 3 lines ]  
^G Помощь  ^O Записать  ^W Поиск     ^K Вырезать  ^J Выводить  ^C ТекПозиц  
^X Выход    ^R ЧитФайл  ^\ Замена    ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.5 – Текст скрипта для задания 3

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.6.

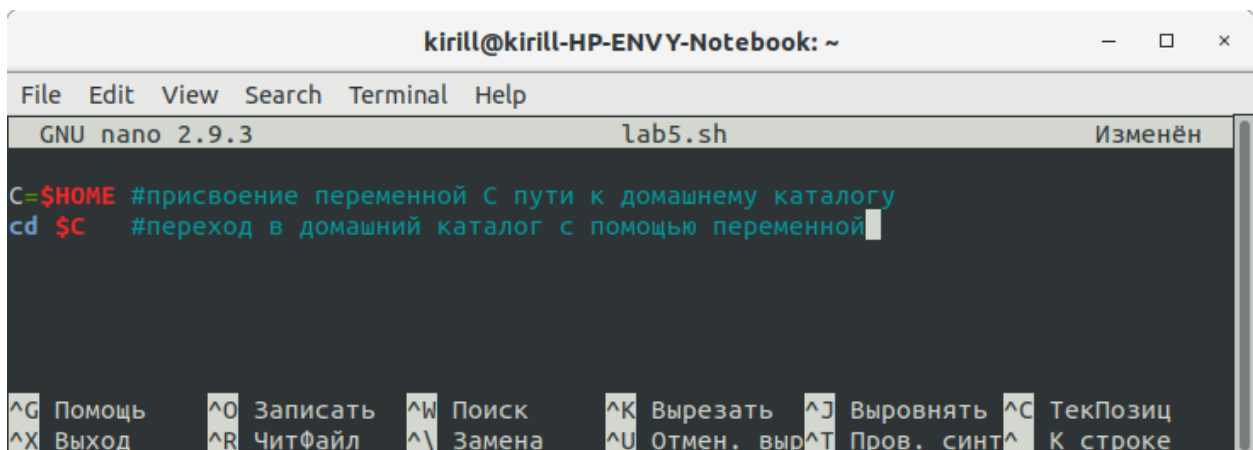


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
10  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.6 – Результат выполнения скрипта для задания 3

4. Присвоить переменной `C` значение "путь до своего каталога". Перейти в этот каталог с использованием переменной.

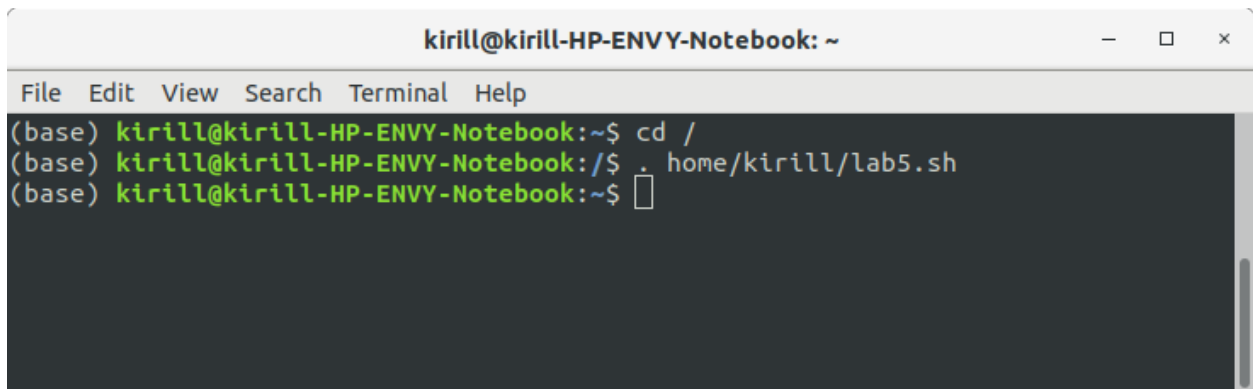
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.7.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh Изменён  
C=$HOME #присвоение переменной C пути к домашнему каталогу  
cd $C #переход в домашний каталог с помощью переменной  
  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.7 – Текст скрипта для задания 4

Перейдем в корневой каталог с помощью команды `cd /`, затем запустим скрипт с помощью команды `.`, чтобы процесс скрипта запустился в текущем экземпляре `bash`. В результате запуска скрипта с помощью команды `.`, получим результат, представленный на рисунке 1.8.

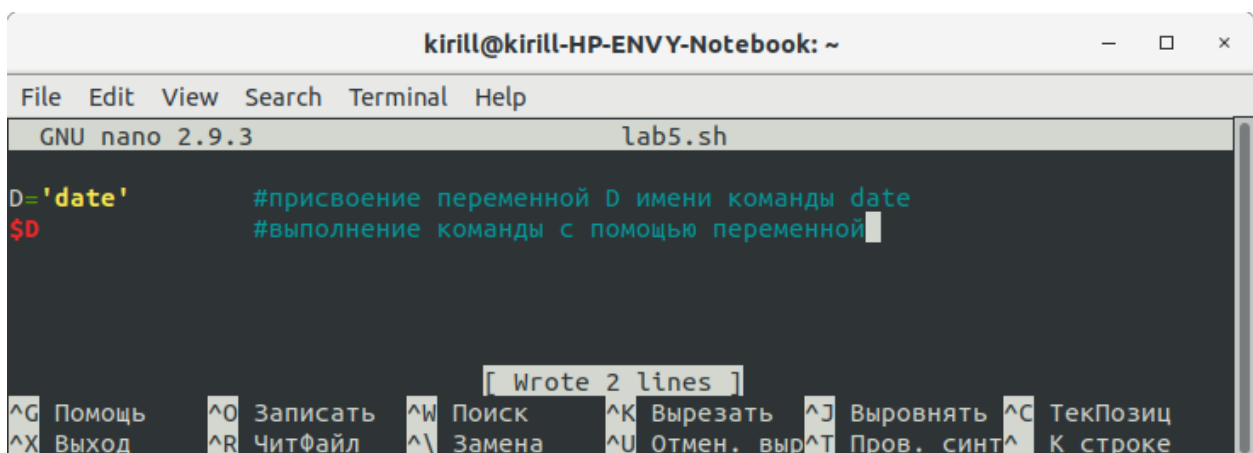


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ cd /  
(base) kirill@kirill-HP-ENVY-Notebook:/$ . home/kirill/lab5.sh  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.8 – Результат выполнения скрипта для задания 4

5. Присвоить переменной D значение “имя команды”, а именно, команды DATE. Выполнить эту команду, используя значение переменной

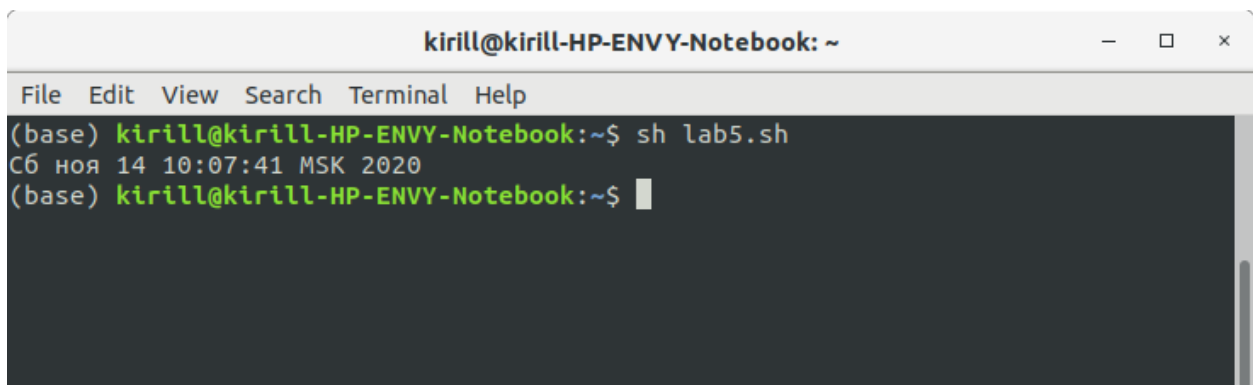
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.9.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh  
D='date'           #присвоение переменной D имени команды date  
$D                #выполнение команды с помощью переменной  
[ Wrote 2 lines ]  
^G Помощь      ^O Записать    ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц  
^X Выход       ^R ЧитФайл    ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.9 – Текст скрипта для задания 3

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.10.

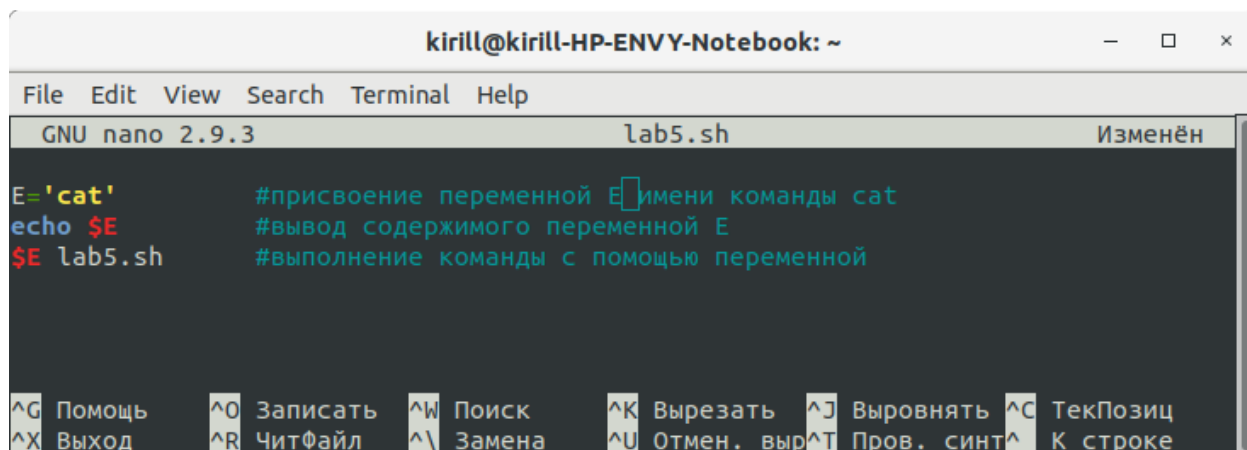


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
Сб ноя 14 10:07:41 MSK 2020  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.10 – Результат выполнения скрипта для задания 5

6. Присвоить переменной E значение “имя команды”, а именно, команды просмотра содержимого файла, просмотреть содержимое переменной. Выполнить эту команду, используя значение переменной

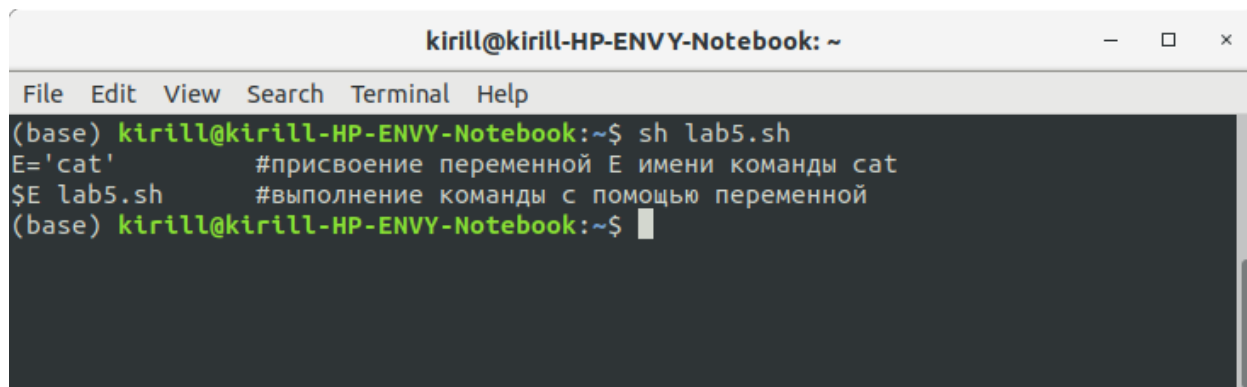
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.11.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён
E='cat'          #присвоение переменной E имени команды cat
echo $E          #вывод содержимого переменной E
$E lab5.sh       #выполнение команды с помощью переменной
^G Помощь      ^O Записать   ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход       ^R ЧитФайл   ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.11 – Текст скрипта для задания 6

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.12.

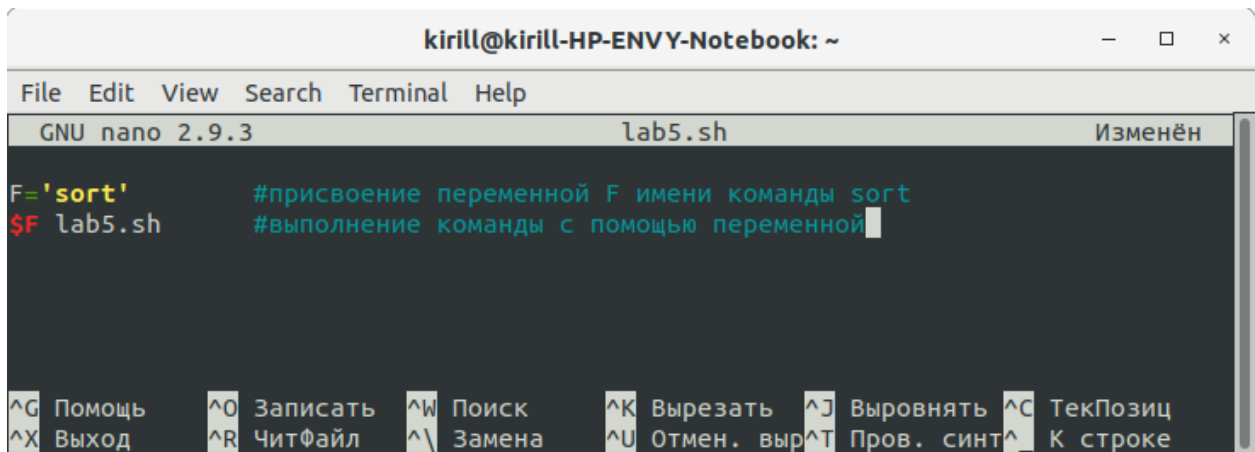


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
E='cat'          #присвоение переменной E имени команды cat
$E lab5.sh       #выполнение команды с помощью переменной
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.12 – Результат выполнения скрипта для задания 6

7. Присвоить переменной F значение “имя команды”, а именно сортировки содержимого текстового файла. Выполнить эту команду, используя значение переменной

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.13.



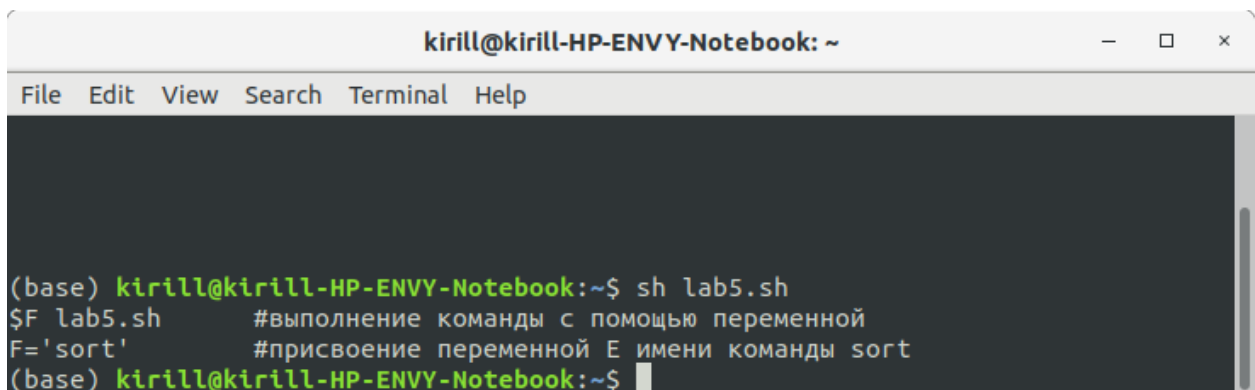
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

F='sort'      #присвоение переменной F имени команды sort
$F lab5.sh    #выполнение команды с помощью переменной

^G Помощь    ^O Записать  ^W Поиск     ^K Вырезать  ^J Выводить  ^C ТекПозиц
^X Выход      ^R ЧитФайл  ^\ Замена    ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.13 – Текст скрипта для задания 7

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.14.



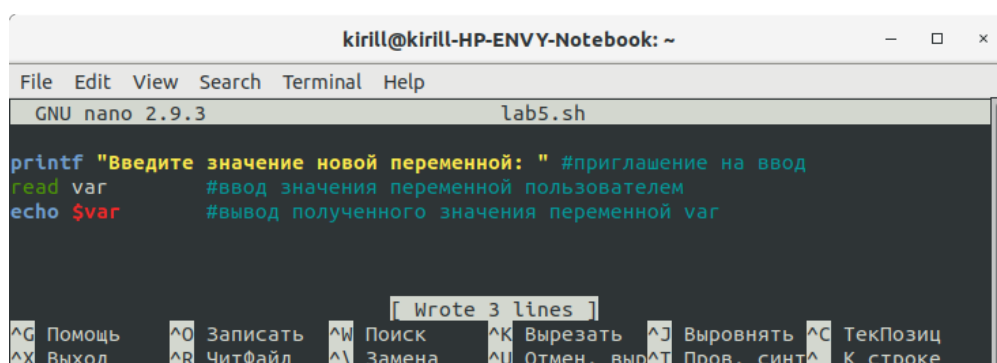
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help

(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
$F lab5.sh      #выполнение команды с помощью переменной
F='sort'       #присвоение переменной E имени команды sort
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.14 – Результат выполнения скрипта для задания 7

8. Программа запрашивает значение переменной, а затем выводит значение этой переменной

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.15.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

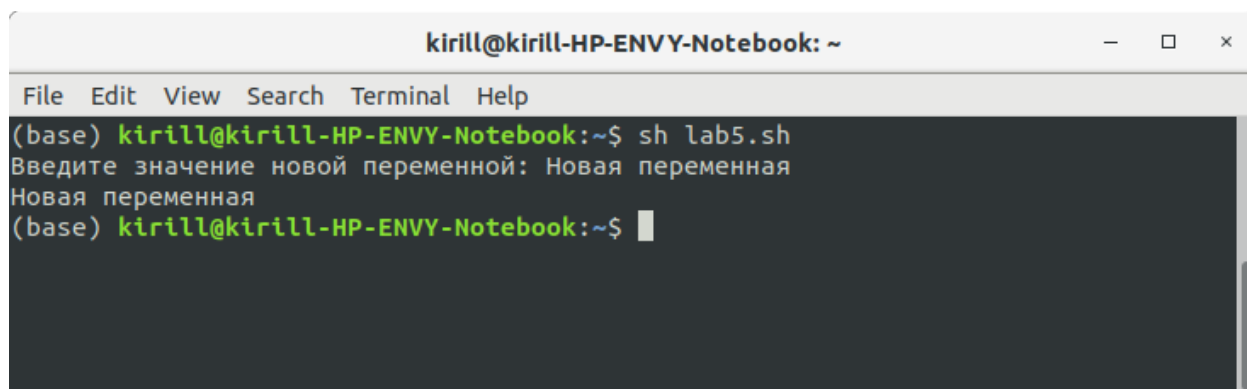
printf "Введите значение новой переменной: " #приглашение на ввод
read var                                     #ввод значения переменной пользователем
echo $var                                   #вывод полученного значения переменной var

[ Wrote 3 lines ]

^G Помощь    ^O Записать  ^W Поиск     ^K Вырезать  ^J Выводить  ^C ТекПозиц
^X Выход      ^R ЧитФайл  ^\ Замена    ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.15 – Текст скрипта для задания 8

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.16.

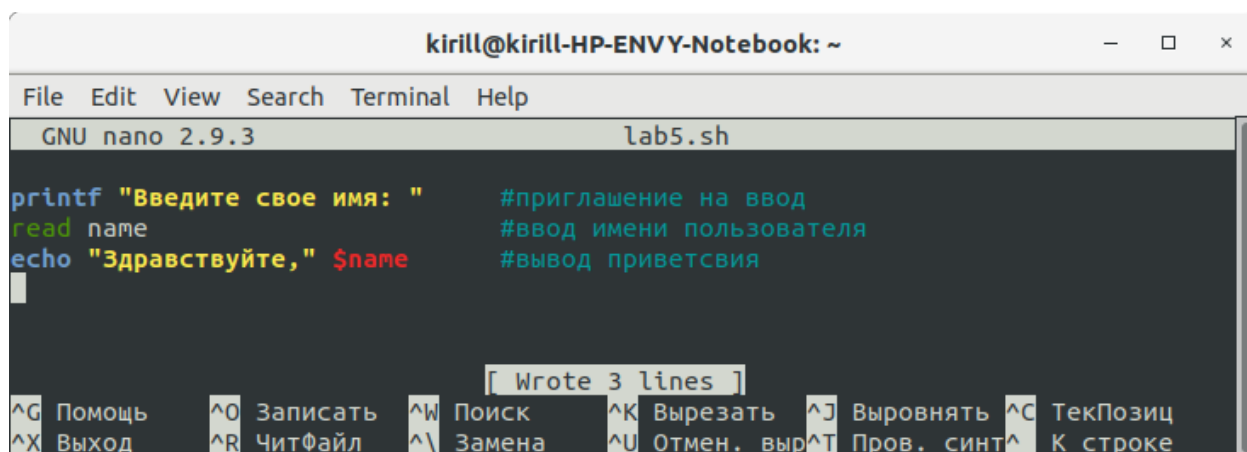


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
Введите значение новой переменной: Новая переменная  
Новая переменная  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.16 – Результат выполнения скрипта для задания 8

9. Программа запрашивает имя пользователя, затем здоровается с ним, используя значение введенной переменной

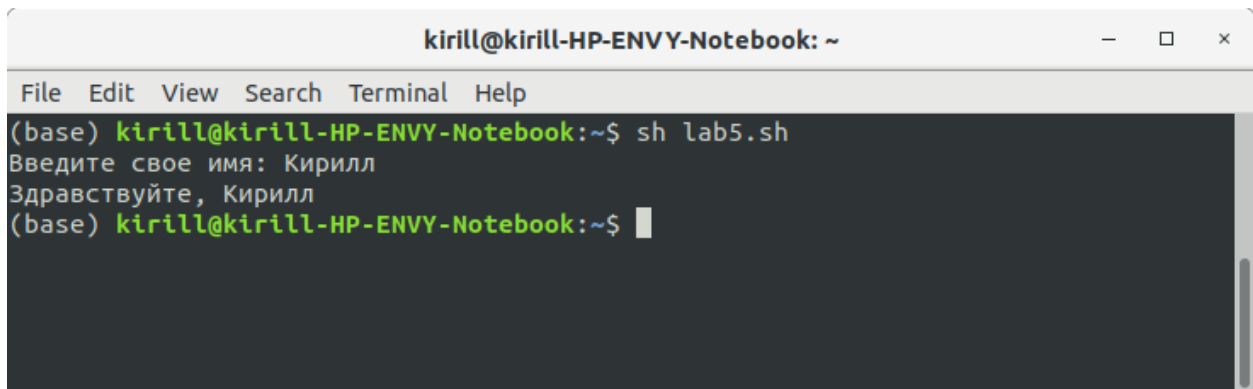
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.17.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh  
printf "Введите свое имя: "      #приглашение на ввод  
read name                       #ввод имени пользователя  
echo "Здравствуйте," $name      #вывод приветствия  
[ Wrote 3 lines ]  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.17 – Текст скрипта для задания 9

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.18.

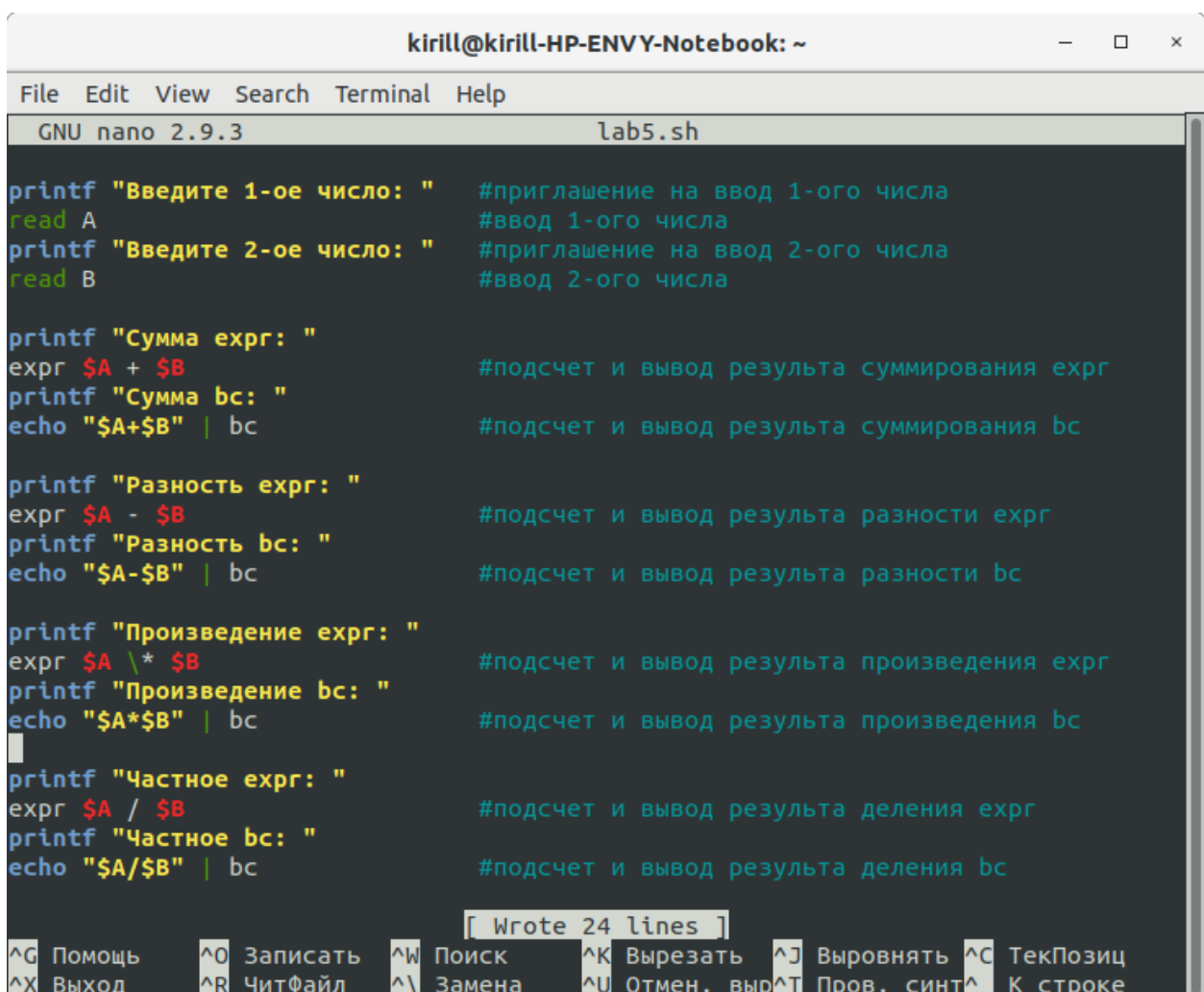


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите свое имя: Кирилл
Здравствуйте, Кирилл
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.18 – Результат выполнения скрипта для задания 9

10. Программа запрашивает значения двух переменных, вычисляет сумму(разность, произведение, деление) этих переменных. Результат выводится на экран (использовать команды а) EXPR; б) BC)

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.19.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

printf "Введите 1-ое число: " #приглашение на ввод 1-ого числа
read A #ввод 1-ого числа
printf "Введите 2-ое число: " #приглашение на ввод 2-ого числа
read B #ввод 2-ого числа

printf "Сумма expr: "
expr $A + $B #подсчет и вывод результата суммирования expr
printf "Сумма bc: "
echo "$A+$B" | bc #подсчет и вывод результата суммирования bc

printf "Разность expr: "
expr $A - $B #подсчет и вывод результата разности expr
printf "Разность bc: "
echo "$A-$B" | bc #подсчет и вывод результата разности bc

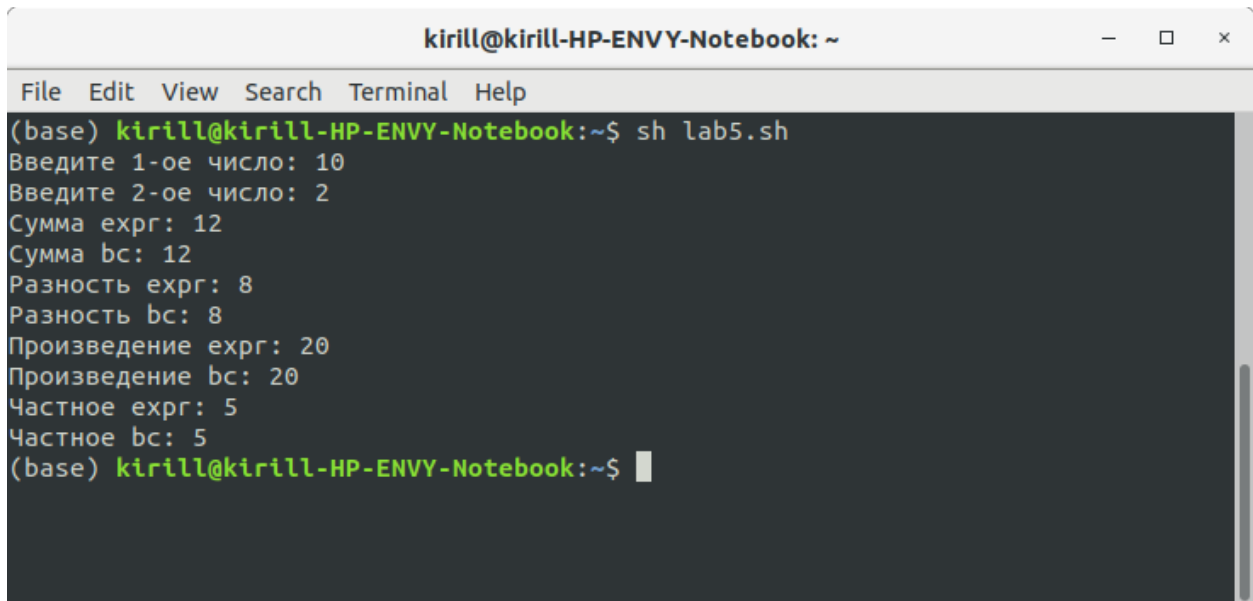
printf "Произведение expr: "
expr $A * $B #подсчет и вывод результата произведения expr
printf "Произведение bc: "
echo "$A*$B" | bc #подсчет и вывод результата произведения bc

printf "Частное expr: "
expr $A / $B #подсчет и вывод результата деления expr
printf "Частное bc: "
echo "$A/$B" | bc #подсчет и вывод результата деления bc

[ Wrote 24 lines ]
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.19 – Текст скрипта для задания 10

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.20.

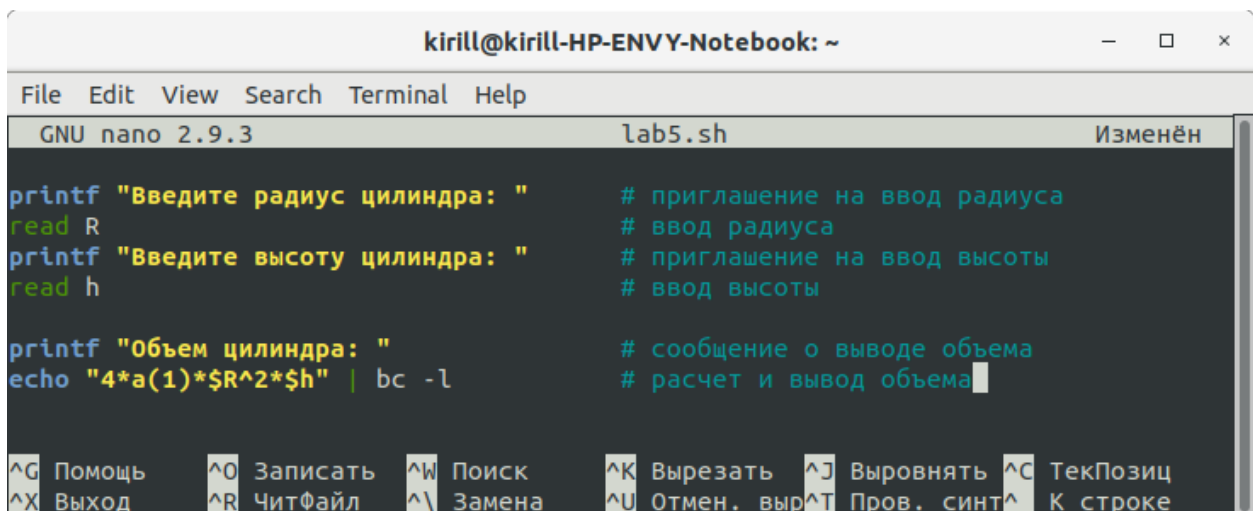


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
Введите 1-ое число: 10  
Введите 2-ое число: 2  
Сумма expr: 12  
Сумма bc: 12  
Разность expr: 8  
Разность bc: 8  
Произведение expr: 20  
Произведение bc: 20  
Частное expr: 5  
Частное bc: 5  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.20 – Результат выполнения скрипта для задания 10

11. Вычислить объем цилиндра. Исходные данные запрашиваются программой. Результат выводится на экран.

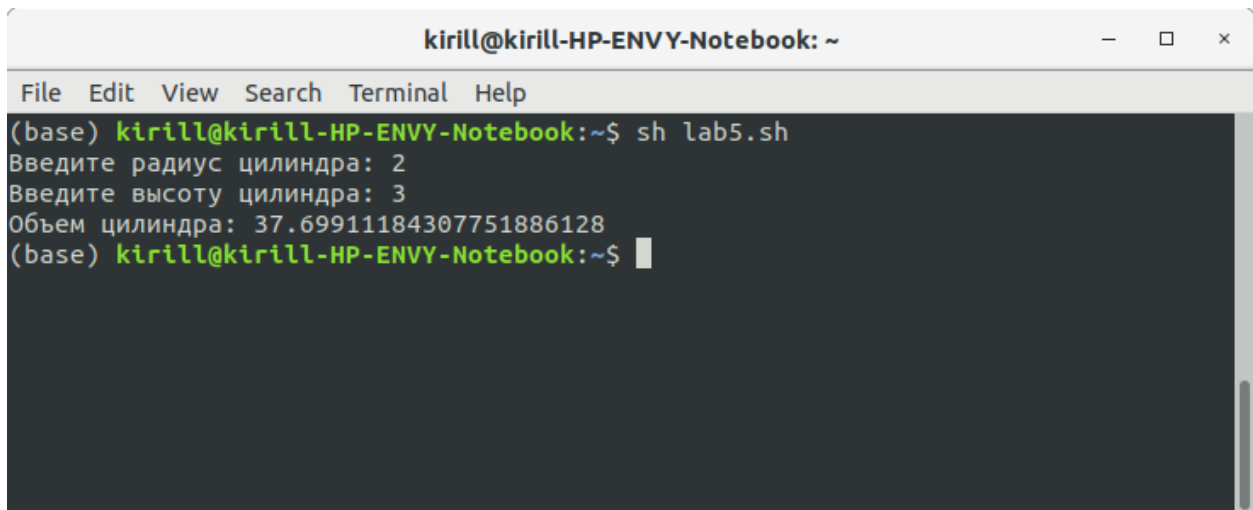
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.21.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh Изменён  
printf "Введите радиус цилиндра: " # приглашение на ввод радиуса  
read R # ввод радиуса  
printf "Введите высоту цилиндра: " # приглашение на ввод высоты  
read h # ввод высоты  
  
printf "Объем цилиндра: " # сообщение о выводе объема  
echo "4*a(1)*$R^2*$h" | bc -l # расчет и вывод объема  
  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.21 – Текст скрипта для задания 11

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.22.

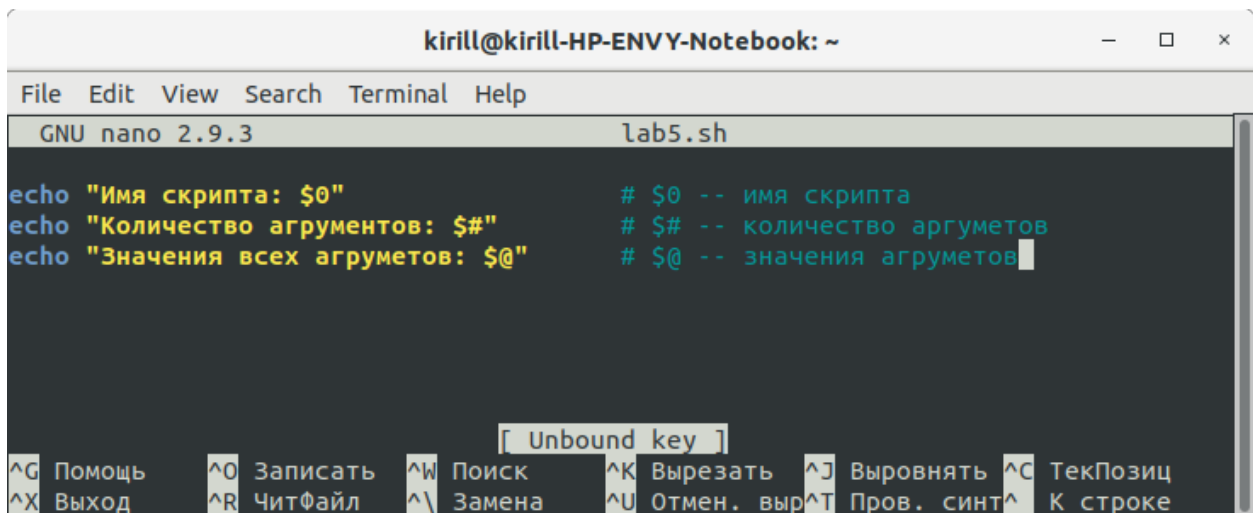


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
Введите радиус цилиндра: 2  
Введите высоту цилиндра: 3  
Объем цилиндра: 37.69911184307751886128  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.22 – Результат выполнения скрипта для задания 11

12. Используя позиционные параметры, отобразить имя программы, количество аргументов командной строки, значение каждого аргумента командной строки.

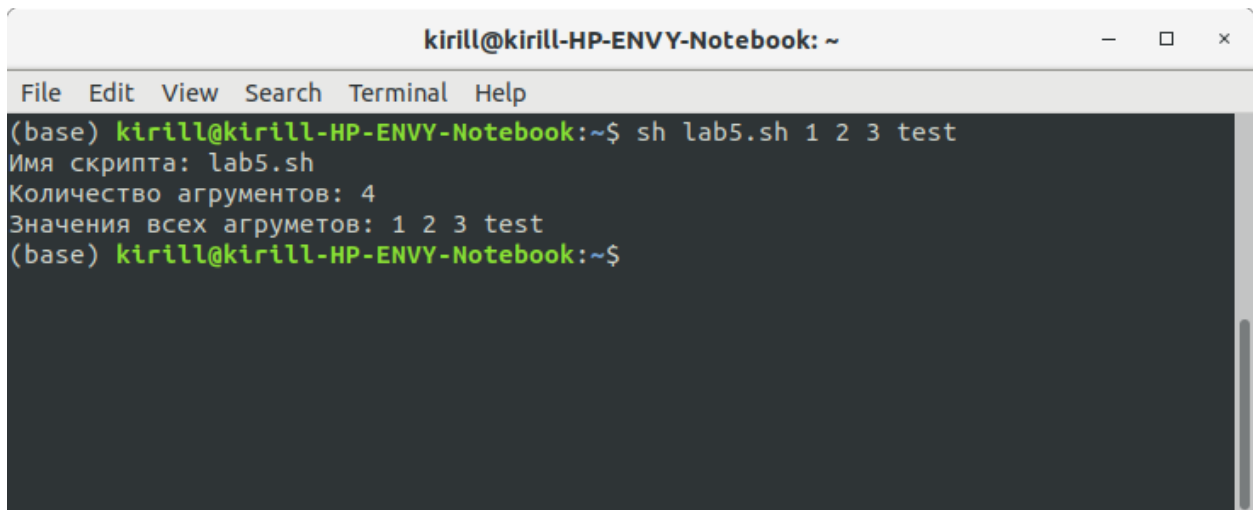
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.23.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh  
echo "Имя скрипта: $0"           # $0 -- имя скрипта  
echo "Количество аргументов: $#" # $# -- количество аргументов  
echo "Значения всех аргументов: $@" # $@ -- значения аргументов  
[ Unbound key ]  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.23 – Текст скрипта для задания 12

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.24.

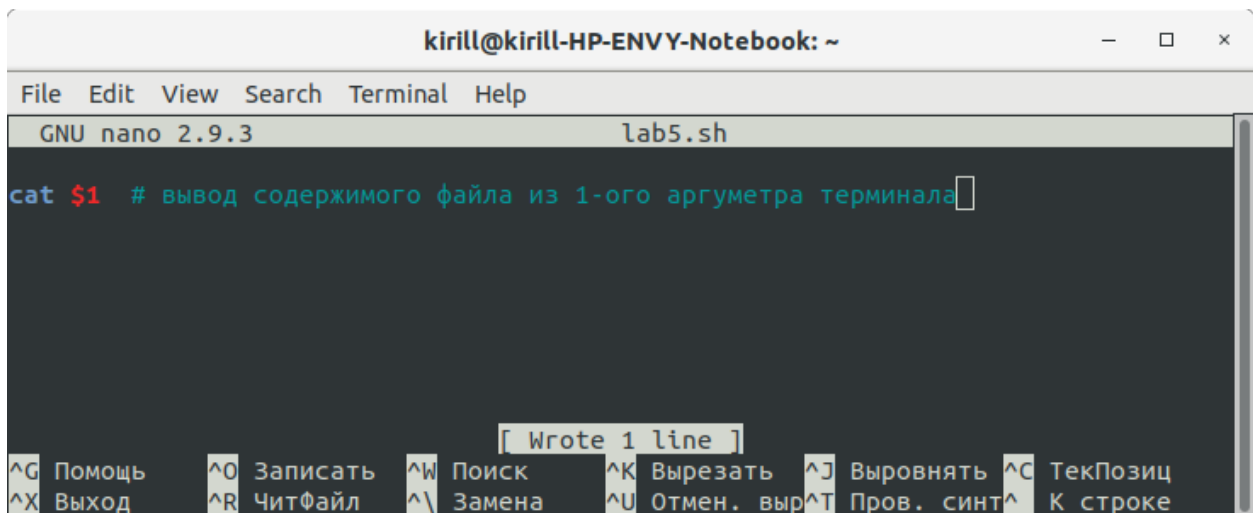


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh 1 2 3 test
Имя скрипта: lab5.sh
Количество аргументов: 4
Значения всех аргументов: 1 2 3 test
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.24 – Результат выполнения скрипта для задания 12

13. Используя позиционный параметр, отобразить содержимое текстового файла, указанного в качестве аргумента командной строки. После паузы экран очищается.

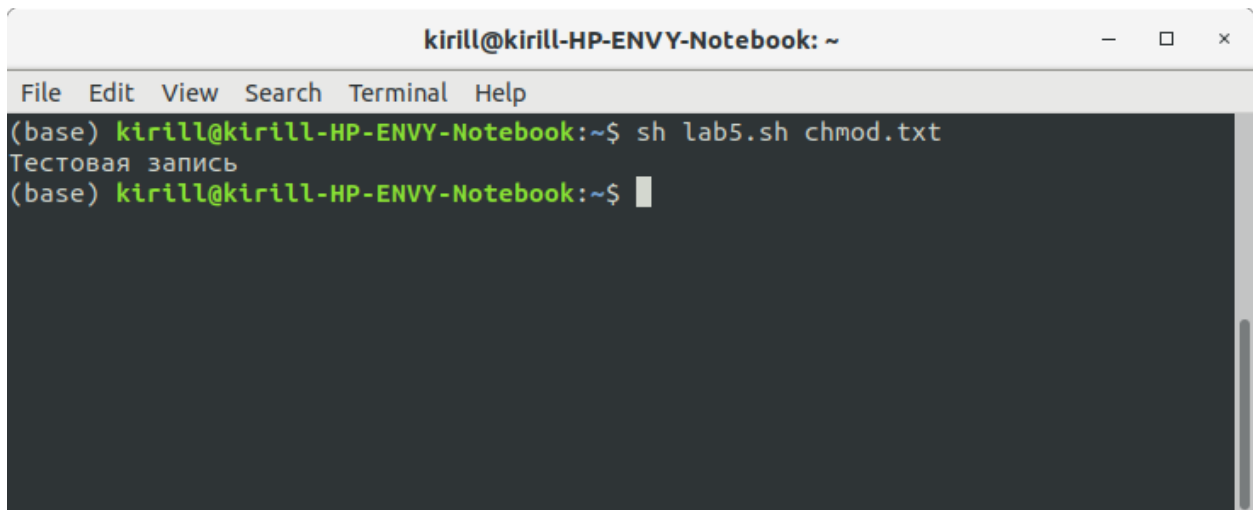
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.25.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh
cat $1 # вывод содержимого файла из 1-ого аргумента терминала
[ Wrote 1 line ]
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.25 – Текст скрипта для задания 13

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.26.

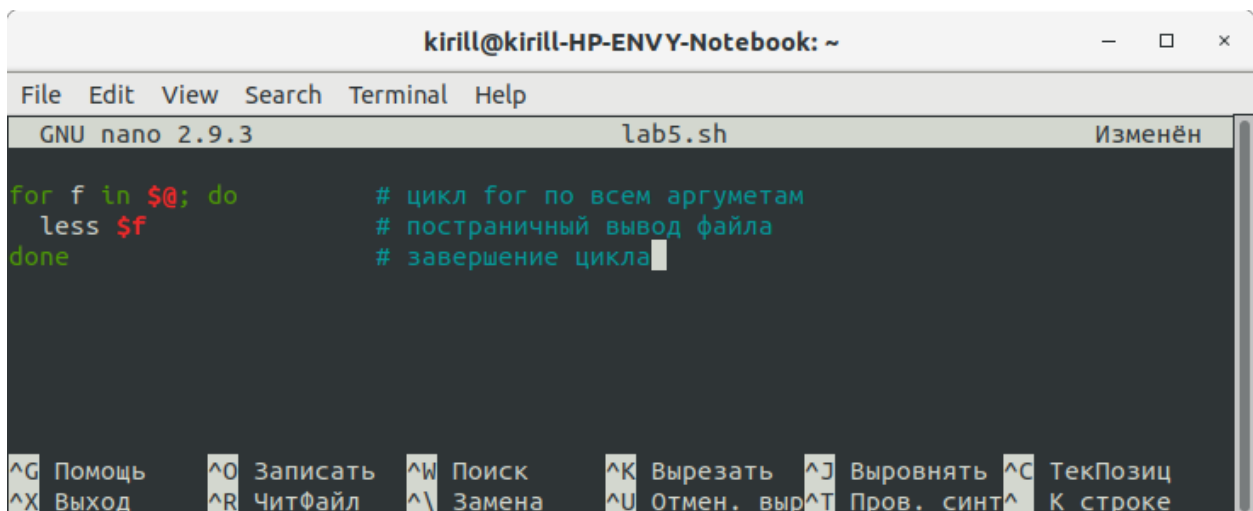


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh chmod.txt
Тестовая запись
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.26 – Результат выполнения скрипта для задания 13

14. Используя оператор FOR, отобразить содержимое текстовых файлов текущего каталога постранично.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.27.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

for f in $@; do          # цикл for по всем аргументам
  less $f                # постраничный вывод файла
done                    # завершение цикла

^G Помощь      ^O Записать   ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход      ^R ЧитФайл   ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.27 – Текст скрипта для задания 14

В результате запуска скрипта с помощью команды `sh lab5.sh top-output.txt test.log`, получим результат, представленный на рисунке 1.28.

```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
top - 01:29:05 up 1:59, 1 user, load average: 1,98, 1,93, 1,82
Tasks: 280 total, 1 running, 227 sleeping, 0 stopped, 0 zombie
%Cpu(s): 21,3 us, 3,7 sy, 0,2 ni, 72,2 id, 1,2 wa, 0,0 hi, 1,4 si, 0,0 st
KiB Mem : 8038836 total, 422336 free, 5560412 used, 2056088 buff/cache
KiB Swap: 8000508 total, 7932992 free, 67516 used. 1675112 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
 3535 kirill    28   8 3137624 423264 162556 S   27,8   5,3   11:27.85 Web Content
 9761 kirill    20   0  45492   4084   3388 R   16,7   0,1    0:00.05 top
   999 gdm       20   0 3901908 158332 99308 S    5,6   2,0    0:19.82 gnome-shell
 1472 kirill    20   0 4166608 342824 105168 S    5,6   4,3    5:41.26 gnome-shell
 1498 kirill    20   0 3171344  17892  13612 S    5,6   0,2    1:45.65 pulseaudio
top-output.txt
```

Рисунок 1.28 – Результат выполнения скрипта для задания 14

15. Программой запрашивается ввод числа, значение которого затем сравнивается с допустимым значением. В результате этого сравнения на экран выдаются соответствующие сообщения.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.29.

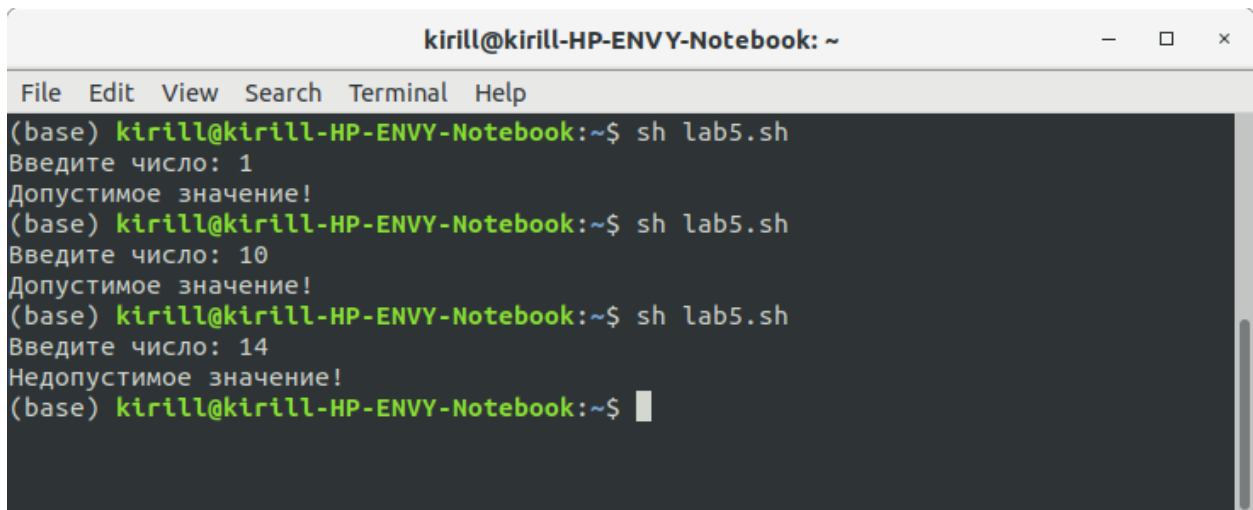
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

printf "Введите число: "          # приглашение на ввод числа
read a                           # ввод числа
if [ $a -gt 10 ];                 # проверка, что число больше 10
then
echo "Недопустимое значение!";   # если да, то сообщение о недопустимости
else
echo "Допустимое значение!";     # если нет, то сообщение о допустимости
fi                                # конец условия

^G Помощь      ^O Записать   ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход       ^R ЧитФайл   ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.29 – Текст скрипта для задания 15

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.30.

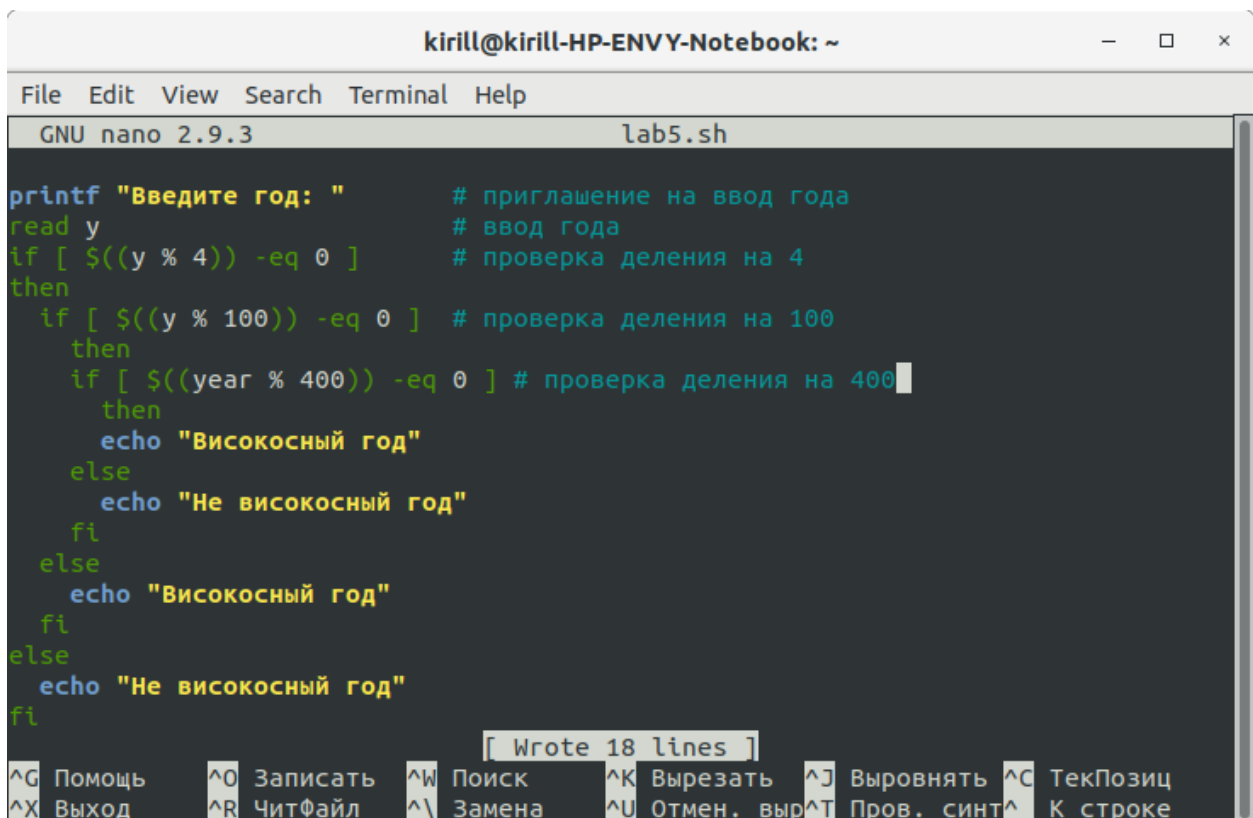


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите число: 1
Допустимое значение!
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите число: 10
Допустимое значение!
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите число: 14
Недопустимое значение!
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.30 – Результат выполнения скрипта для задания 15

16. Программой запрашивается год, определяется, високосный ли он. Результат выдается на экран.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.31.



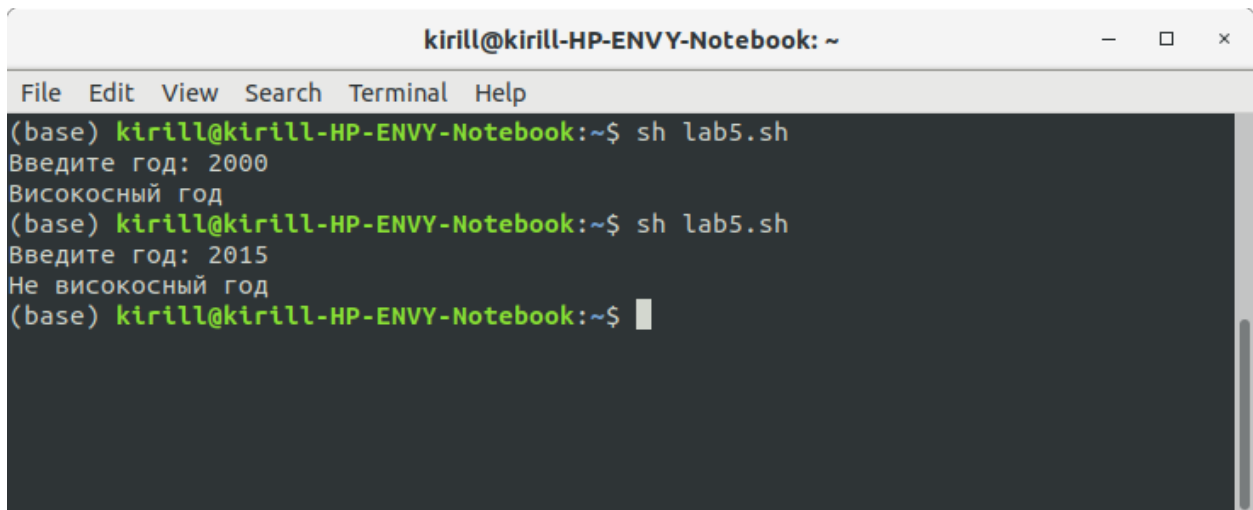
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

printf "Введите год: "      # приглашение на ввод года
read y                     # ввод года
if [  $$(y \% 4)$  -eq 0 ]      # проверка деления на 4
then
    if [  $$(y \% 100)$  -eq 0 ]  # проверка деления на 100
    then
        if [  $$(year \% 400)$  -eq 0 ] # проверка деления на 400
        then
            echo "Високосный год"
        else
            echo "Не високосный год"
        fi
    else
        echo "Високосный год"
    fi
else
    echo "Не високосный год"
fi

[ Wrote 18 lines ]
^G Помощь  ^O Записать  ^W Поиск    ^K Вырезать ^J Выводить  ^C ТекПозиц
^X Выход    ^R ЧитФайл  ^\ Замена  ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.31 – Текст скрипта для задания 16

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.32.

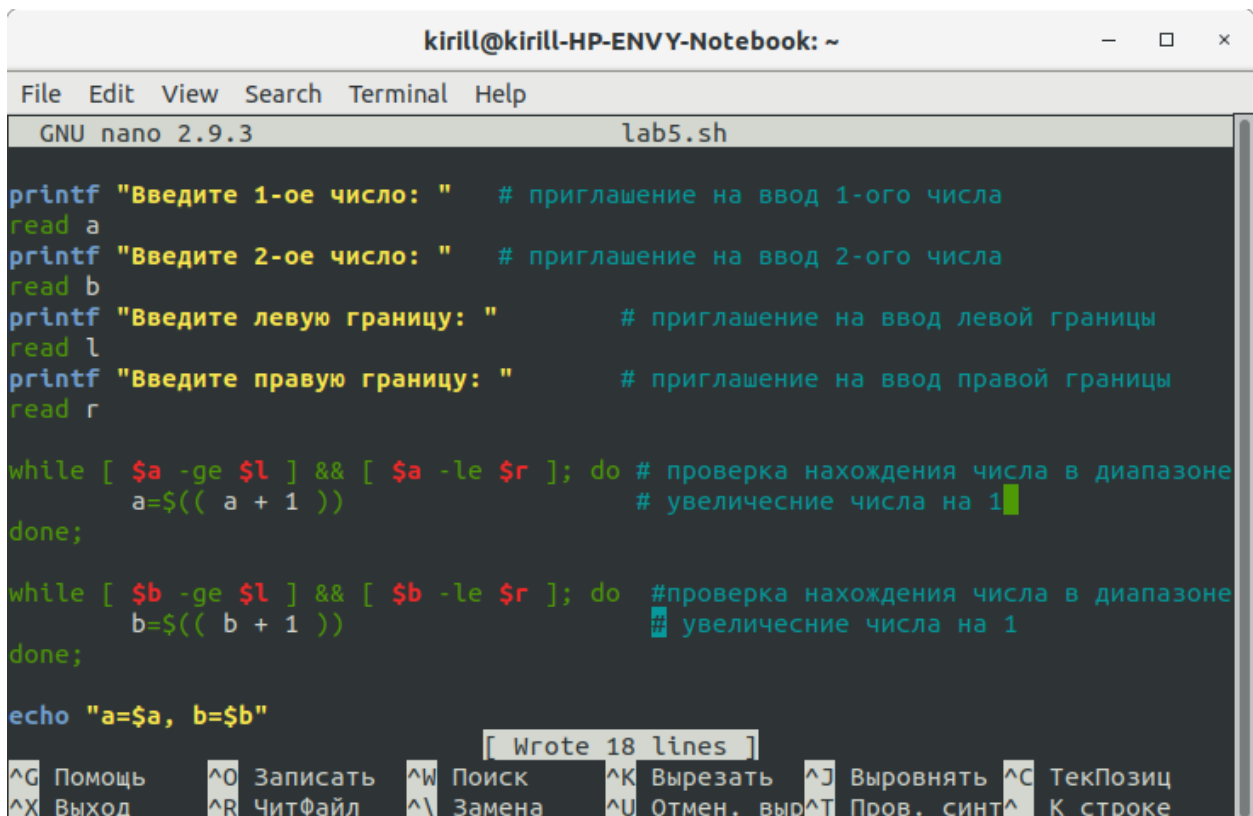


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите год: 2000
Високосный год
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите год: 2015
Не високосный год
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.32 – Результат выполнения скрипта для задания 16

17. Вводятся целочисленные значения двух переменных. Вводится диапазон данных. Пока значения переменных находятся в указанном диапазоне, их значения инкрементируются.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.33.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

printf "Введите 1-ое число: " # приглашение на ввод 1-ого числа
read a
printf "Введите 2-ое число: " # приглашение на ввод 2-ого числа
read b
printf "Введите левую границу: " # приглашение на ввод левой границы
read l
printf "Введите правую границу: " # приглашение на ввод правой границы
read r

while [ $a -ge $l ] && [ $a -le $r ]; do # проверка нахождения числа в диапазоне
    a=$(( a + 1 )) # увеличание числа на 1
done;

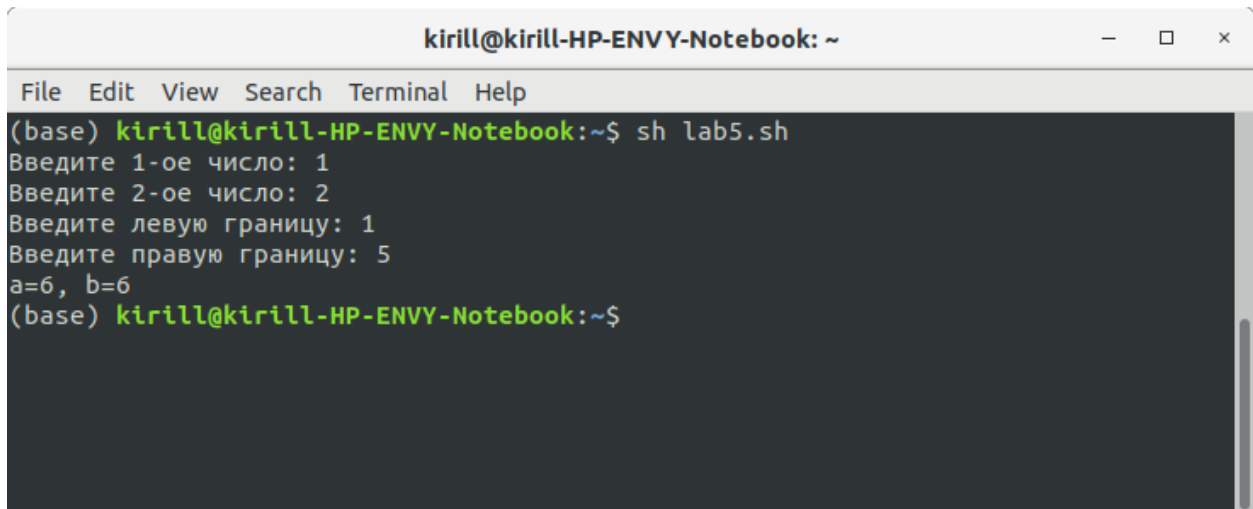
while [ $b -ge $l ] && [ $b -le $r ]; do # проверка нахождения числа в диапазоне
    b=$(( b + 1 )) # увеличание числа на 1
done;

echo "a=$a, b=$b"

[ Wrote 18 lines ]
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.33 – Текст скрипта для задания 17

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.34.

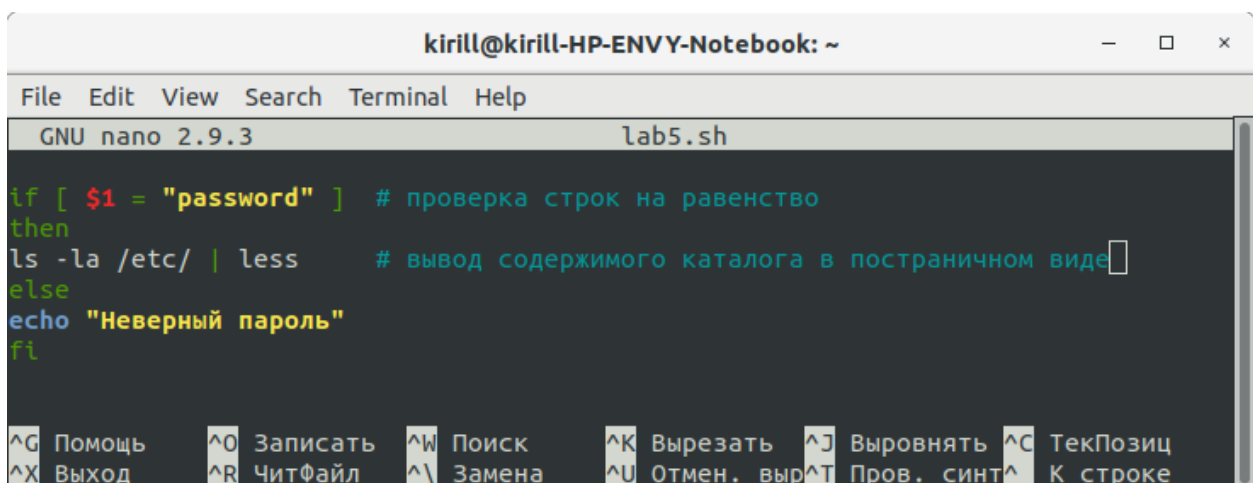


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
Введите 1-ое число: 1
Введите 2-ое число: 2
Введите левую границу: 1
Введите правую границу: 5
a=6, b=6
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.34 – Результат выполнения скрипта для задания 17

18. В качестве аргумента командной строки указывается пароль. Если пароль введен верно, постранично отображается в длинном формате с указанием скрытых файлов содержимое каталога `/etc`.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.35.



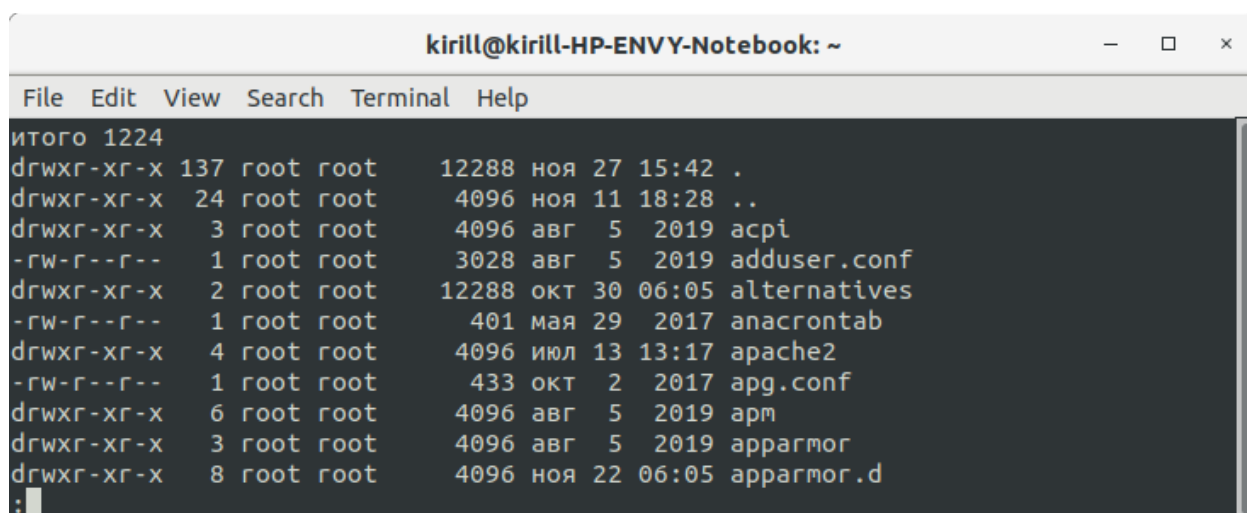
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

if [ $1 = "password" ] # проверка строк на равенство
then
ls -la /etc/ | less    # вывод содержимого каталога в постраничном виде
else
echo "Неверный пароль"
fi

^G Помощь      ^O Записать    ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход      ^R ЧитФайл    ^\ Замена     ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.35 – Текст скрипта для задания 18

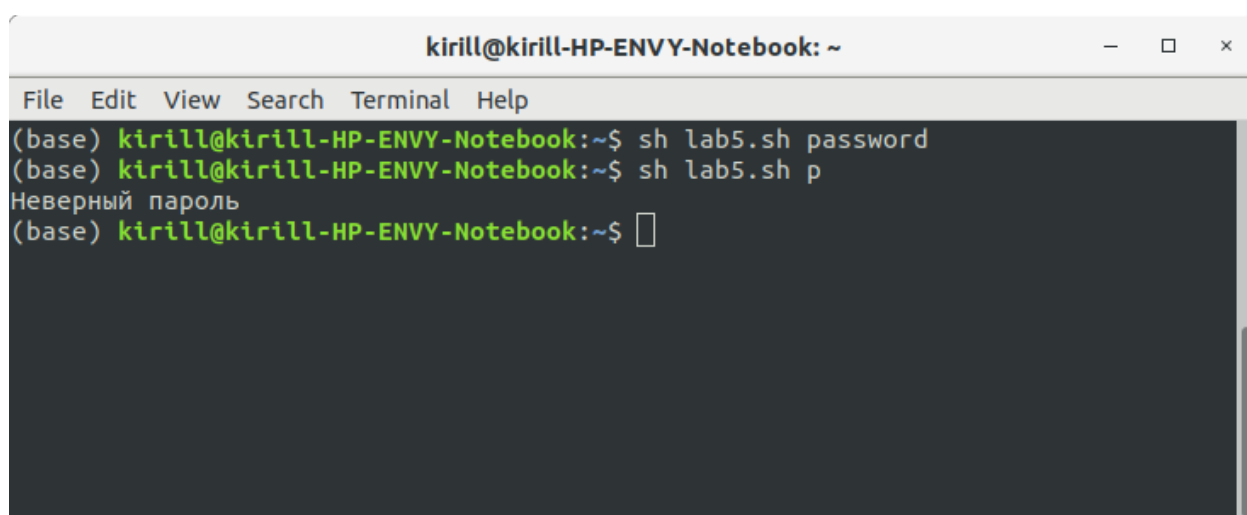
В результате запуска скрипта с помощью команды `sh lab5.sh password`, получим результат, представленный на рисунке 1.36.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
итого 1224
drwxr-xr-x 137 root root 12288 ноя 27 15:42 .
drwxr-xr-x 24 root root 4096 ноя 11 18:28 ..
drwxr-xr-x 3 root root 4096 авг 5 2019 acpi
-rw-r--r-- 1 root root 3028 авг 5 2019 adduser.conf
drwxr-xr-x 2 root root 12288 окт 30 06:05 alternatives
-rw-r--r-- 1 root root 401 мая 29 2017 anacrontab
drwxr-xr-x 4 root root 4096 июл 13 13:17 apache2
-rw-r--r-- 1 root root 433 окт 2 2017 apg.conf
drwxr-xr-x 6 root root 4096 авг 5 2019 apm
drwxr-xr-x 3 root root 4096 авг 5 2019 apparmor
drwxr-xr-x 8 root root 4096 ноя 22 06:05 apparmor.d
:
```

Рисунок 1.36 – Результат выполнения скрипта для задания 18

В результате запуска скрипта с помощью команды `sh lab5.sh p`, получим результат, представленный на рисунке 1.37.

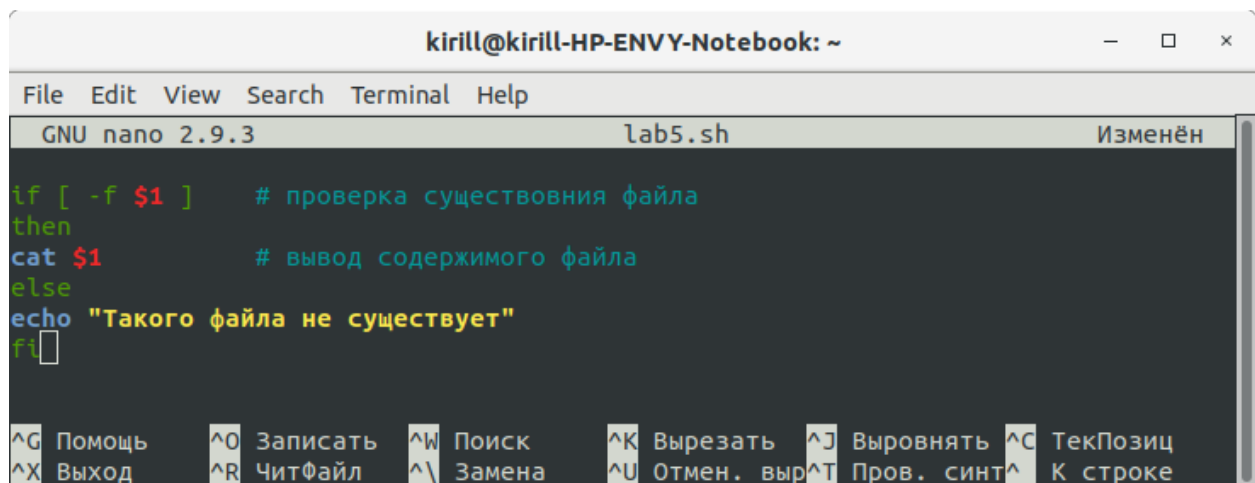


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh password
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh p
Неверный пароль
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.37 – Результат выполнения скрипта для задания 18

19. Проверить, существует ли файл. Если да, выводится на экран его содержимое, если нет - выдается соответствующее сообщение.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.38.

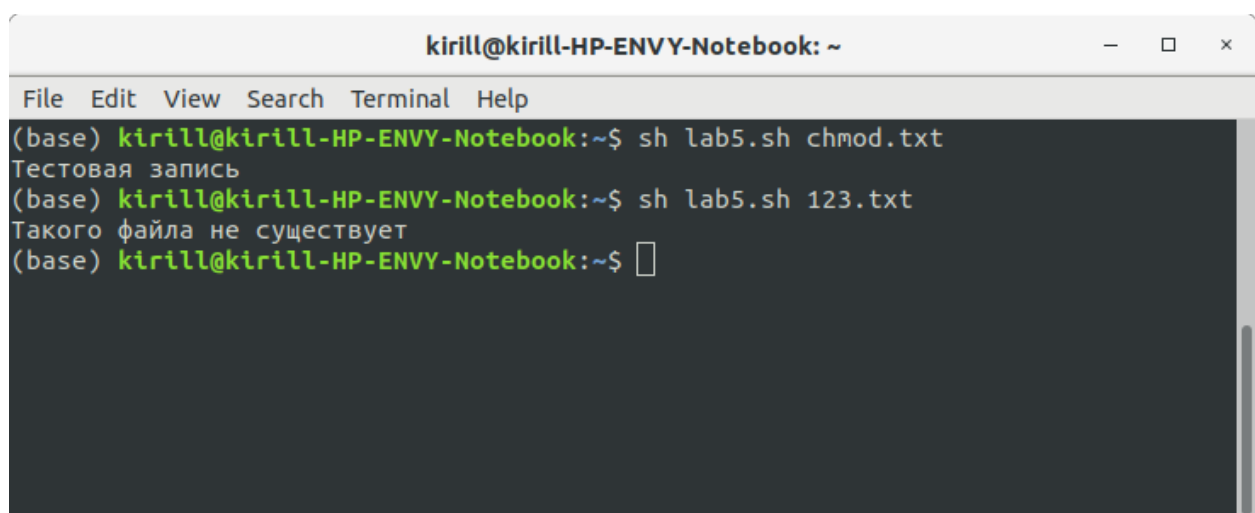


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

if [ -f $1 ]      # проверка существования файла
then
cat $1           # вывод содержимого файла
else
echo "Такого файла не существует"
fi
```

Рисунок 1.38 – Текст скрипта для задания 19

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.39.

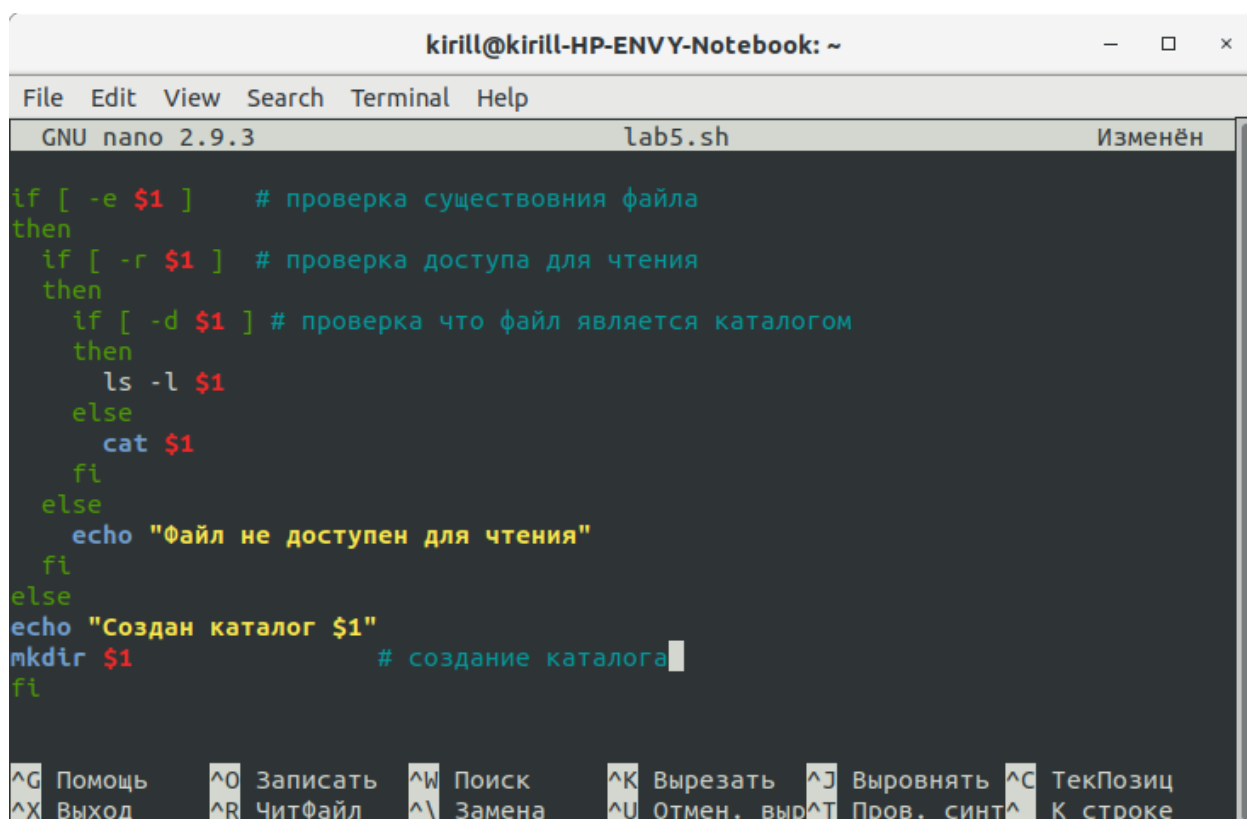


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh chmod.txt
Тестовая запись
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh 123.txt
Такого файла не существует
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.39 – Результат выполнения скрипта для задания 19

20. Если файл есть каталог и этот каталог можно читать, просматривается содержимое этого каталога. Если каталог отсутствует, он создается. Если файл не есть каталог, просматривается содержимое файла.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.40.



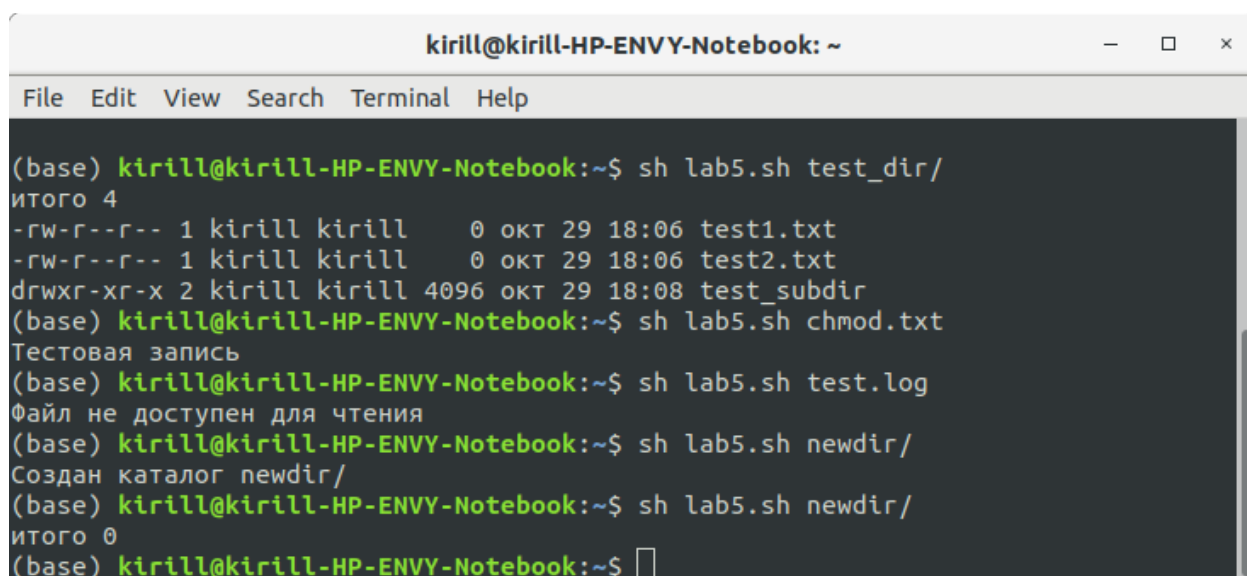
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

if [ -e $1 ] # проверка существования файла
then
  if [ -r $1 ] # проверка доступа для чтения
  then
    if [ -d $1 ] # проверка что файл является каталогом
    then
      ls -l $1
    else
      cat $1
    fi
  else
    echo "Файл не доступен для чтения"
  fi
else
  echo "Создан каталог $1"
  mkdir $1 # создание каталога
fi

^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выровнять ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.40 – Текст скрипта для задания 20

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.41.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help

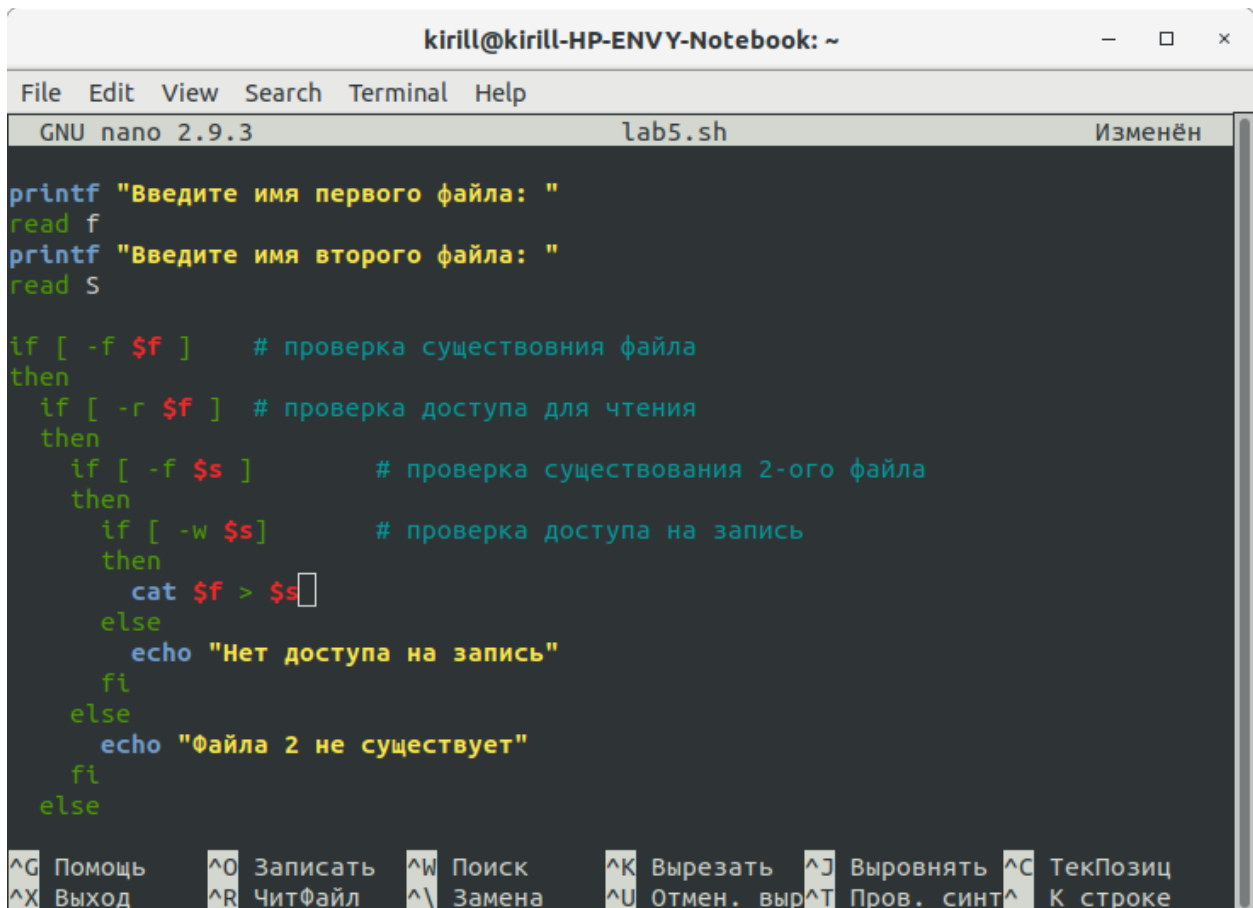
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh test_dir/
итого 4
-rw-r--r-- 1 kirill kirill 0 окт 29 18:06 test1.txt
-rw-r--r-- 1 kirill kirill 0 окт 29 18:06 test2.txt
drwxr-xr-x 2 kirill kirill 4096 окт 29 18:08 test_subdir
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh chmod.txt
Тестовая запись
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh test.log
Файл не доступен для чтения
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh newdir/
Создан каталог newdir/
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh newdir/
итого 0
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.41 – Результат выполнения скрипта для задания 20

21. Анализируются атрибуты файла. Если первый файл существует и используется для чтения, а второй файл существует и используется для записи, то содержимое первого файла перенаправляется во второй файл.

В случае несовпадений указанных атрибутов или отсутствия файлов на экран выдаются соответствующие сообщения (использовать а) имена файлов; б) позиционные параметры).

Текст скрипта для пункта а) с комментариями последовательности действий представлен на рисунке 1.42.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh Изменён

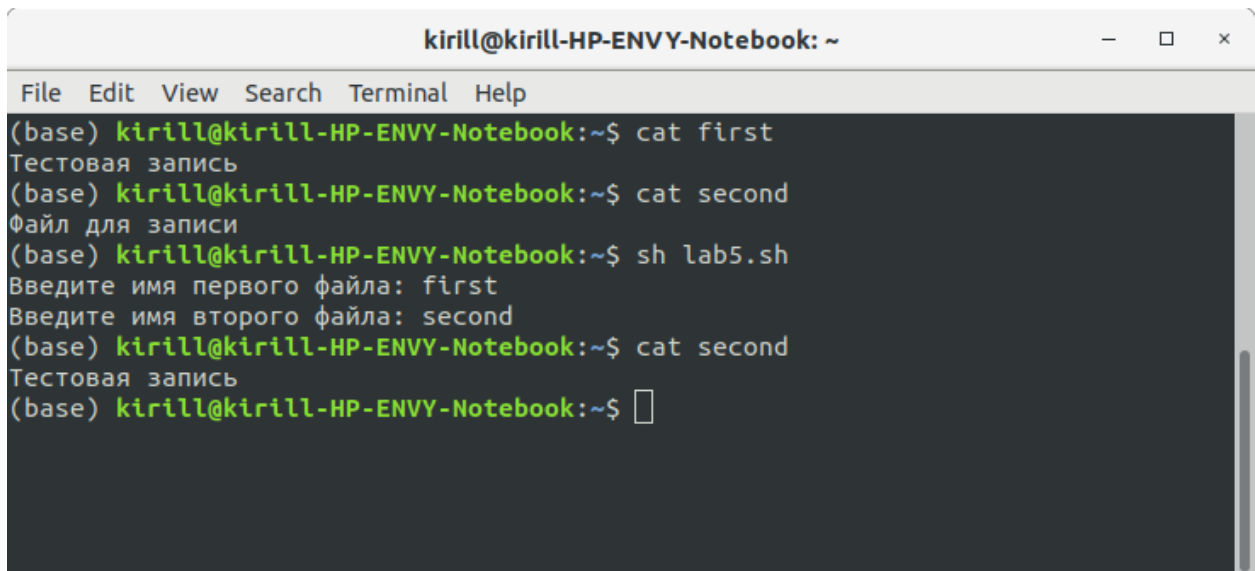
printf "Введите имя первого файла: "
read f
printf "Введите имя второго файла: "
read s

if [ -f $f ] # проверка существования файла
then
  if [ -r $f ] # проверка доступа для чтения
  then
    if [ -f $s ] # проверка существования 2-ого файла
    then
      if [ -w $s ] # проверка доступа на запись
      then
        cat $f > $s
      else
        echo "Нет доступа на запись"
      fi
    else
      echo "Файла 2 не существует"
    fi
  else
  fi
else
  echo "Файла 1 не существует"
fi

^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выровнять ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.42 – Текст скрипта для задания 21 пункт а)

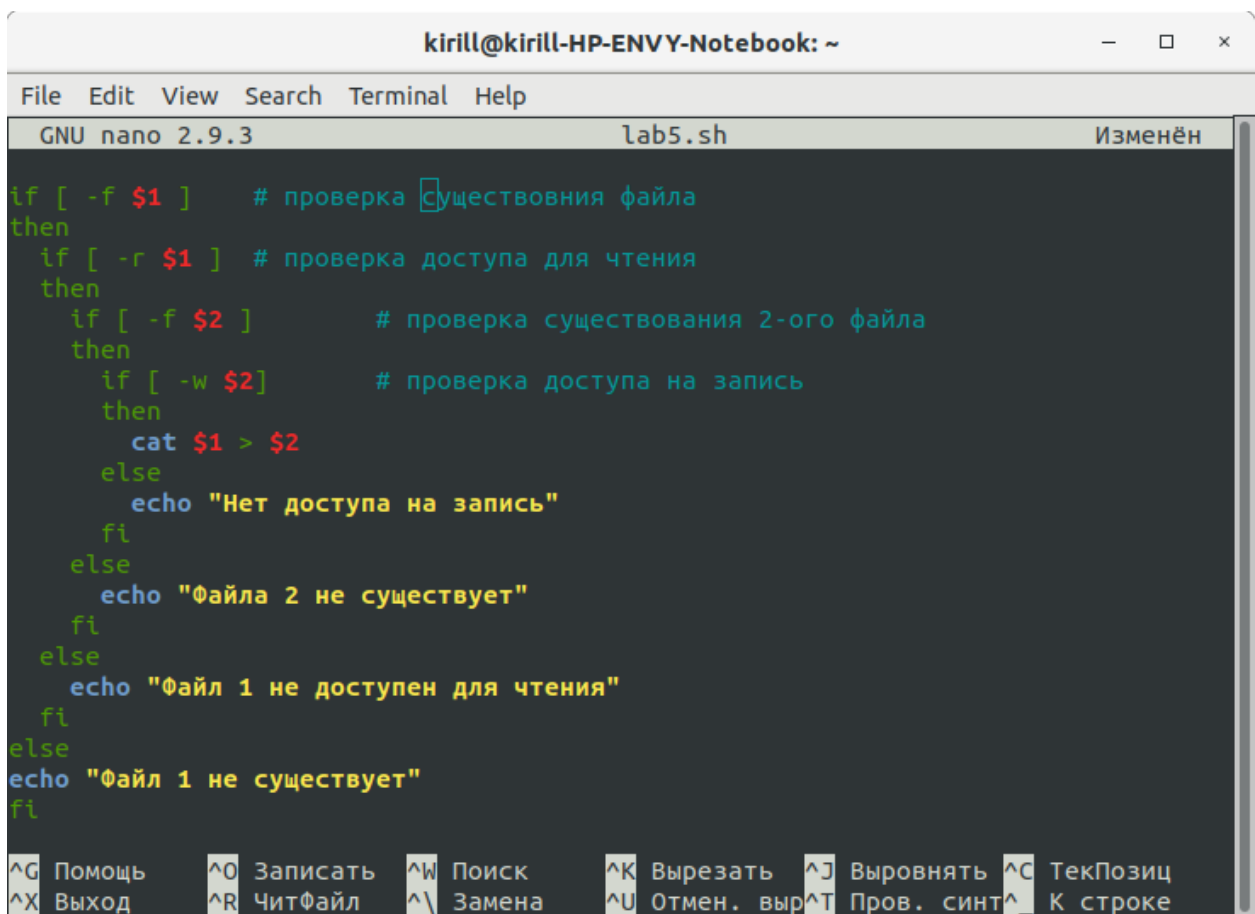
В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.43.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat first  
Тестовая запись  
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat second  
Файл для записи  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh  
Введите имя первого файла: first  
Введите имя второго файла: second  
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat second  
Тестовая запись  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.43 – Результат выполнения скрипта для задания 21 пункт а)

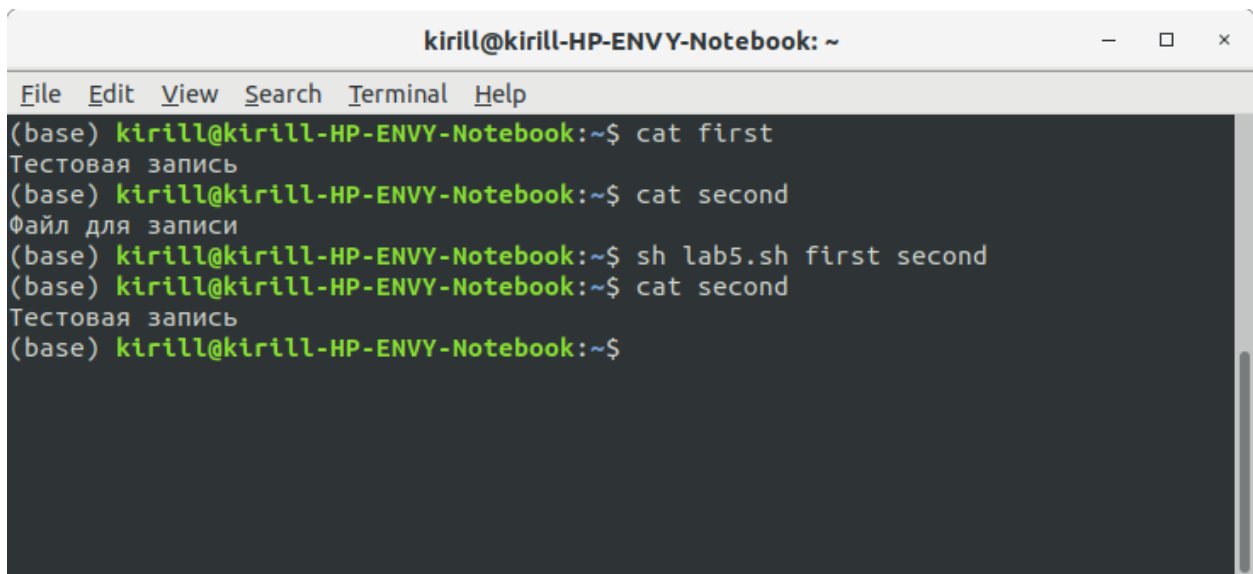
Текст скрипта для пункта б) с комментариями последовательности действий представлен на рисунке 1.44.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh Изменён  
if [ -f $1 ] # проверка существования файла  
then  
  if [ -r $1 ] # проверка доступа для чтения  
  then  
    if [ -f $2 ] # проверка существования 2-ого файла  
    then  
      if [ -w $2 ] # проверка доступа на запись  
      then  
        cat $1 > $2  
      else  
        echo "Нет доступа на запись"  
      fi  
    else  
      echo "Файла 2 не существует"  
    fi  
  else  
    echo "Файл 1 не доступен для чтения"  
  fi  
else  
  echo "Файл 1 не существует"  
fi  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.44 – Текст скрипта для задания 21 пункт б)

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.43.

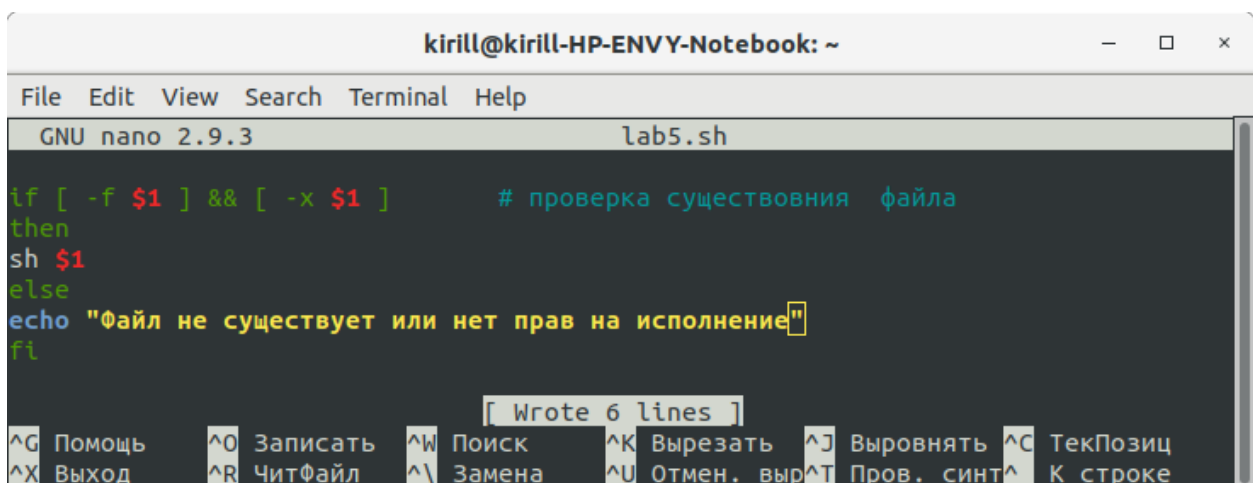


```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat first
Тестовая запись
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat second
Файл для записи
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh first second
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat second
Тестовая запись
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.45 – Результат выполнения скрипта для задания 21 пункт б)

22. Если файл запуска программы найден, программа запускается (по выбору).

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.46.



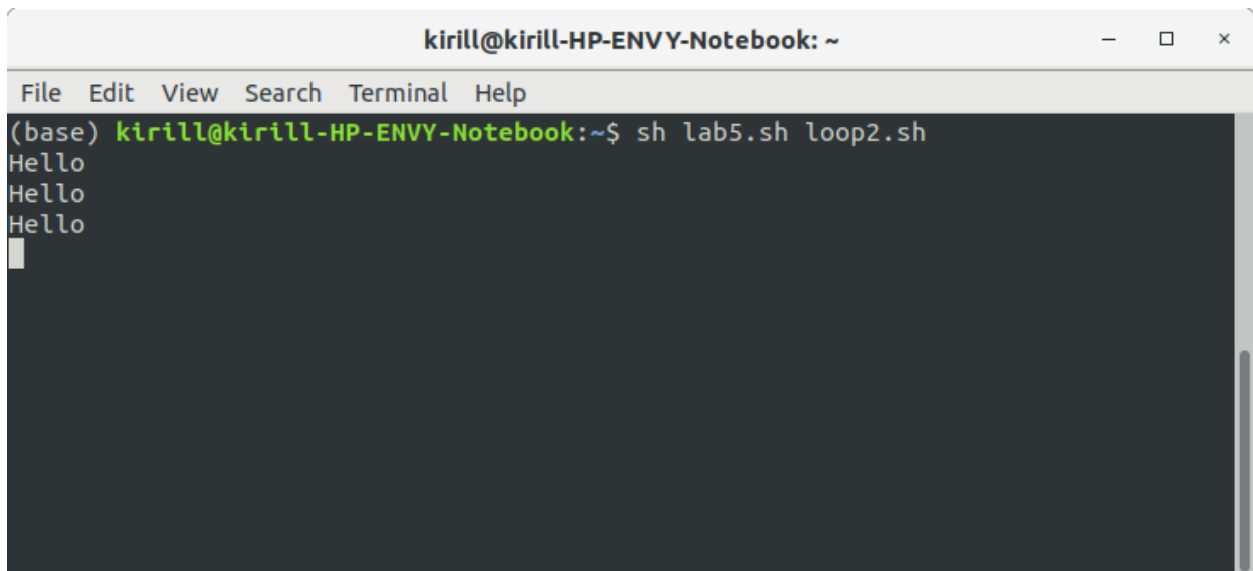
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

if [ -f $1 ] && [ -x $1 ]      # проверка существования файла
then
sh $1
else
echo "Файл не существует или нет прав на исполнение"
fi

[ Wrote 6 lines ]
^G Помощь  ^O Записать  ^W Поиск    ^K Вырезать  ^J Выводить  ^C ТекПозиц
^X Выход    ^R ЧитФайл  ^\ Замена  ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.46 – Текст скрипта для задания 22

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.47.

A terminal window titled "kirill@kirill-HP-ENVY-Notebook: ~" with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is "(base) kirill@kirill-HP-ENVY-Notebook:~\$". The user has entered the command "sh lab5.sh loop2.sh". The output shows three lines of "Hello" stacked vertically, with a cursor on the line following the last "Hello".

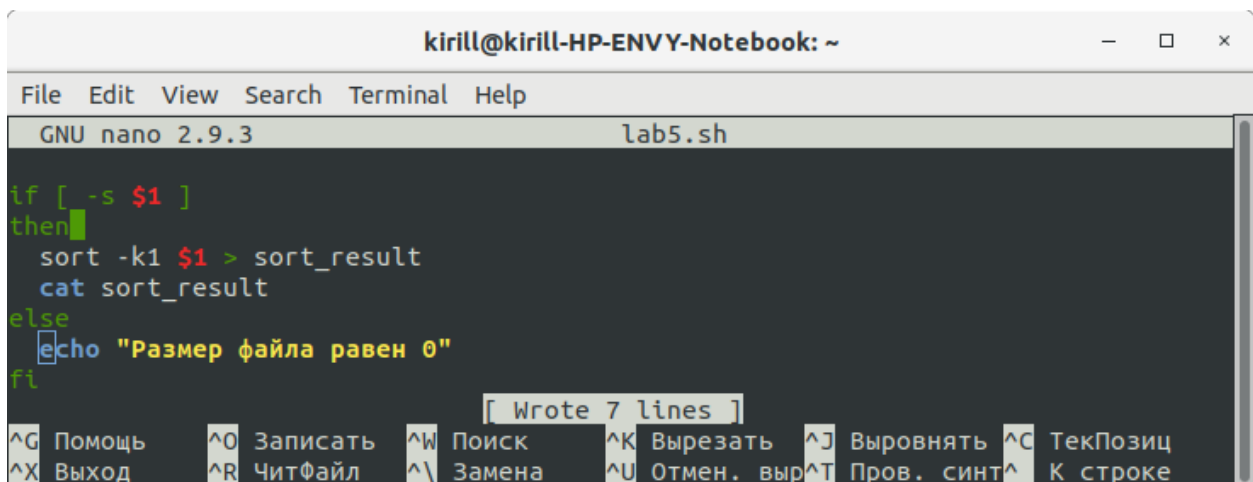
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh loop2.sh
Hello
Hello
Hello

```

Рисунок 1.47 – Результат выполнения скрипта для задания 22

23. В качестве позиционного параметра задается файл, анализируется его размер. Если размер файла больше нуля, содержимое файла сортируется по первому столбцу по возрастанию, отсортированная информация помещается в другой файл, содержимое которого затем отображается на экране.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.48.

A terminal window titled "kirill@kirill-HP-ENVY-Notebook: ~" showing the GNU nano 2.9.3 editor editing "lab5.sh". The script content is visible, including an if-then-else-fi block. A status bar at the bottom shows "Wrote 7 lines" and various keyboard shortcuts in Russian.

```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

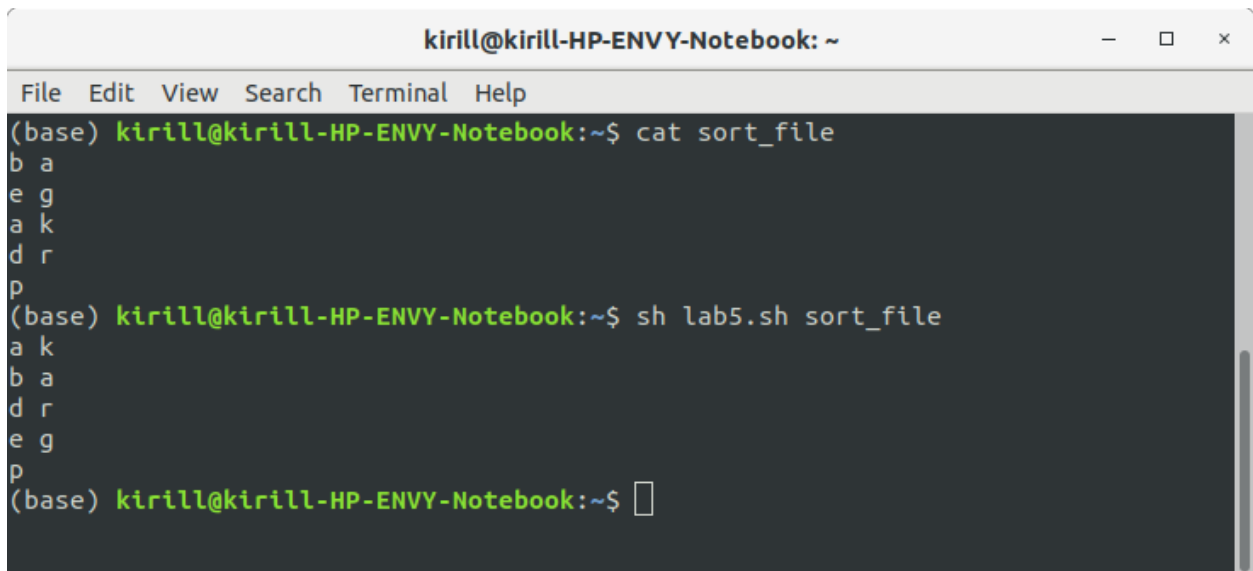
if [ -s $1 ]
then
    sort -k1 $1 > sort_result
    cat sort_result
else
    echo "Размер файла равен 0"
fi

[Wrote 7 lines]
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке

```

Рисунок 1.48 – Текст скрипта для задания 23

В результате запуска скрипта с помощью команды sh, получим результат, представленный на рисунке 1.49.

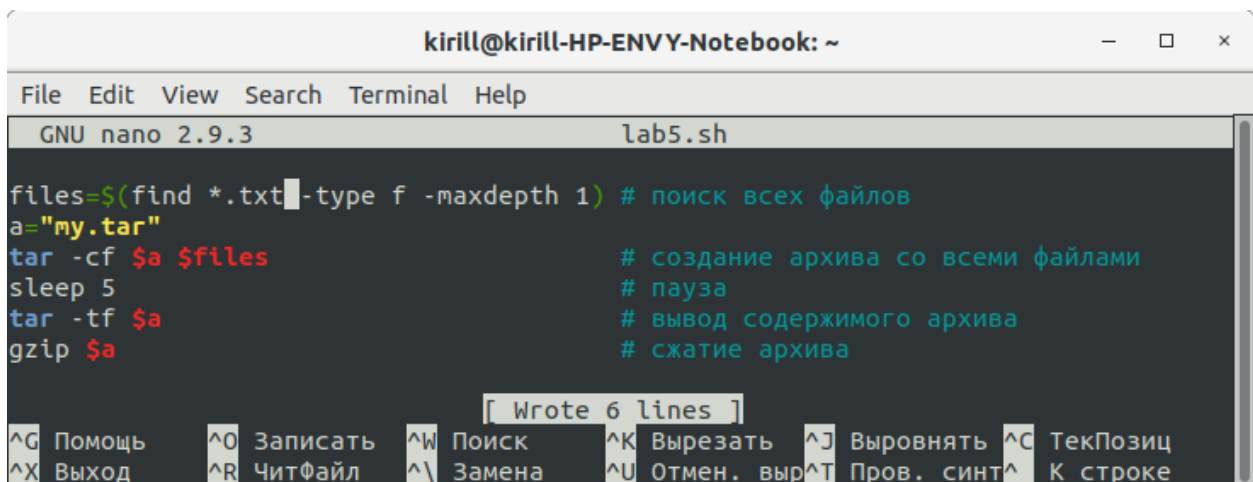


```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
(base) kirill@kirill-HP-ENVY-Notebook:~$ cat sort_file  
b a  
e g  
a k  
d r  
p  
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh sort_file  
a k  
b a  
d r  
e g  
p  
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.49 – Результат выполнения скрипта для задания 23

24. Командой TAR осуществляется сборка всех текстовых файлов текущего каталога в один архивный файл `my.tar`, после паузы просматривается содержимое файла `my.tar`, затем командой GZIP архивный файл `my.tar` сжимается.

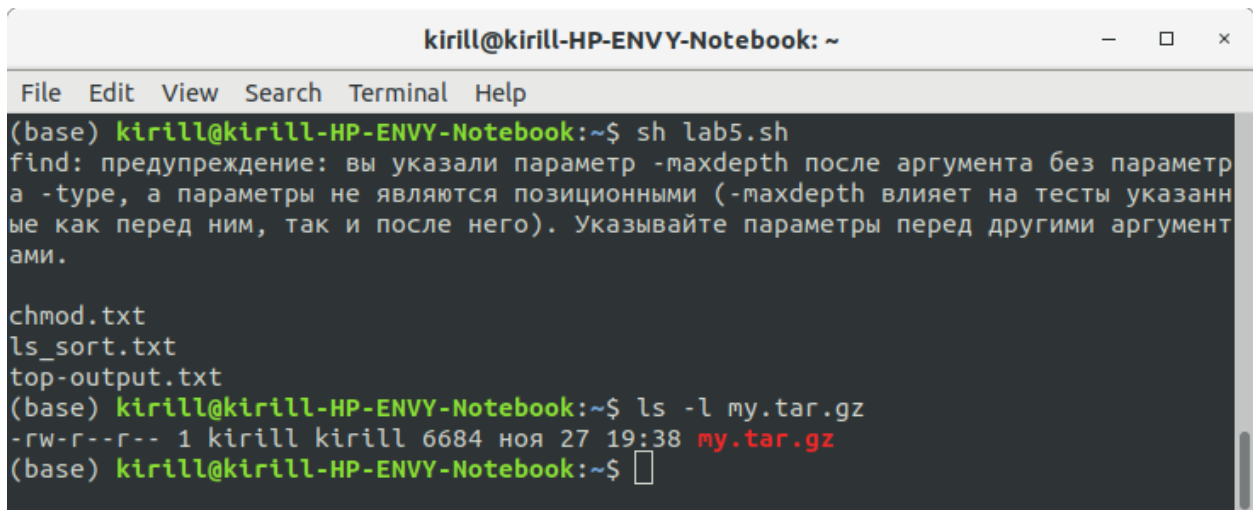
Текст скрипта с комментариями последовательности действий представлен на рисунке 1.50.



```
kirill@kirill-HP-ENVY-Notebook: ~  
File Edit View Search Terminal Help  
GNU nano 2.9.3 lab5.sh  
files=$(find *.txt -type f -maxdepth 1) # поиск всех файлов  
a="my.tar"  
tar -cf $a $files # создание архива со всеми файлами  
sleep 5 # пауза  
tar -tf $a # вывод содержимого архива  
gzip $a # сжатие архива  
[ Wrote 6 lines ]  
^G Помощь ^O Записать ^W Поиск ^K Вырезать ^J Выводить ^C ТекПозиц  
^X Выход ^R ЧитФайл ^\ Замена ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.50 – Текст скрипта для задания 24

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.51.



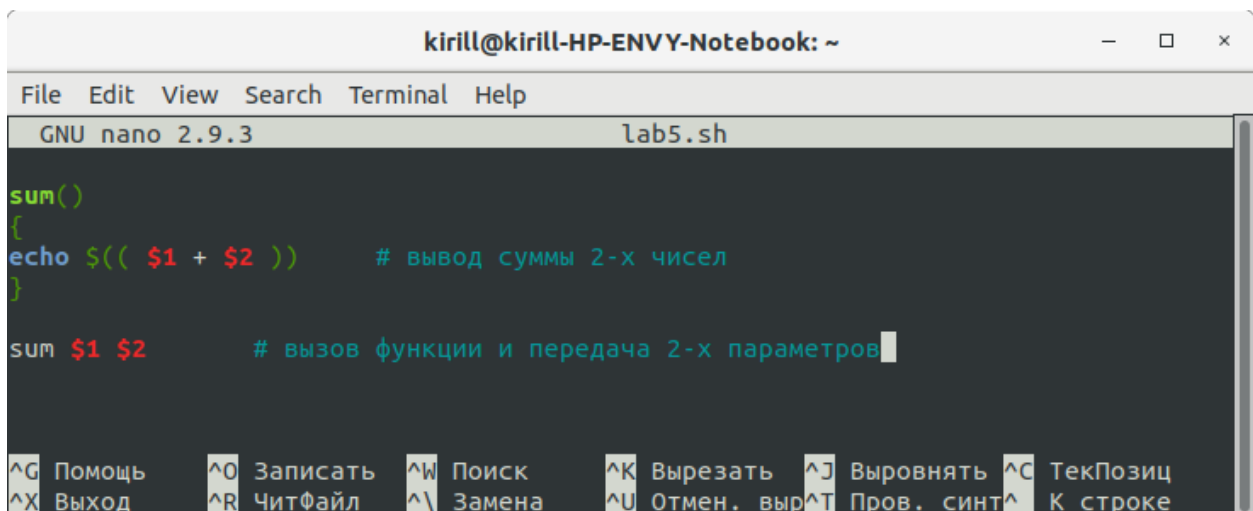
```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh
find: предупреждение: вы указали параметр -maxdepth после аргумента без параметр
а -type, а параметры не являются позиционными (-maxdepth влияет на тесты указанн
ые как перед ним, так и после него). Указывайте параметры перед другими аргумент
ами.

chmod.txt
ls_sort.txt
top-output.txt
(base) kirill@kirill-HP-ENVY-Notebook:~$ ls -l my.tar.gz
-rw-r--r-- 1 kirill kirill 6684 ноя 27 19:38 my.tar.gz
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.51 – Результат выполнения скрипта для задания 24

25. Написать скрипт с использованием функции, например, функции, суммирующей значения двух переменных.

Текст скрипта с комментариями последовательности действий представлен на рисунке 1.52.



```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
GNU nano 2.9.3 lab5.sh

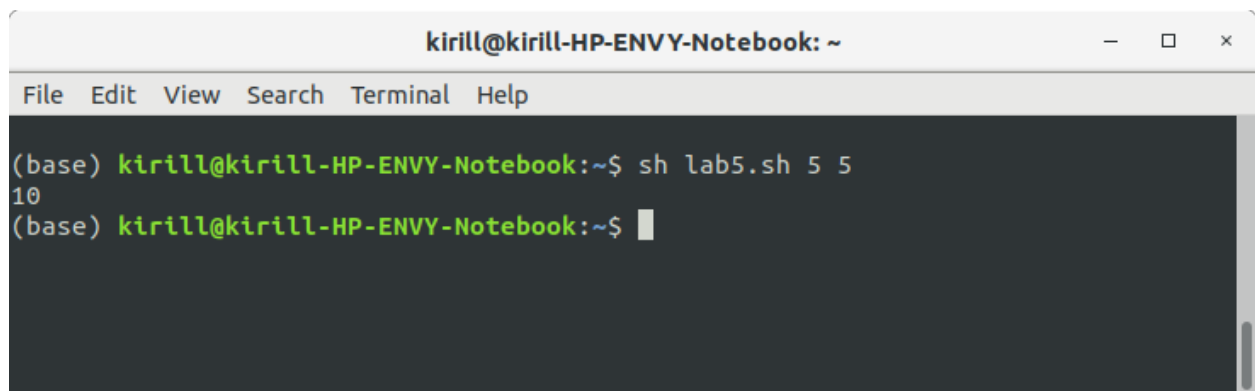
sum()
{
echo $(( $1 + $2 ))      # вывод суммы 2-х чисел
}

sum $1 $2                # вызов функции и передача 2-х параметров

^G Помощь      ^O Записать    ^W Поиск      ^K Вырезать   ^J Выводить   ^C ТекПозиц
^X Выход      ^R ЧитФайл    ^\ Замена    ^U Отмен. выр ^T Пров. синт ^_ К строке
```

Рисунок 1.52 – Текст скрипта для задания 25

В результате запуска скрипта с помощью команды `sh`, получим результат, представленный на рисунке 1.53.

A terminal window titled 'kirill@kirill-HP-ENVY-Notebook: ~' with a menu bar containing 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal output shows a prompt '(base) kirill@kirill-HP-ENVY-Notebook:~\$' followed by the command 'sh lab5.sh 5 5'. The output of the script is '10'. The prompt then changes to '(base) kirill@kirill-HP-ENVY-Notebook:~\$' with a cursor.

```
kirill@kirill-HP-ENVY-Notebook: ~
File Edit View Search Terminal Help
(base) kirill@kirill-HP-ENVY-Notebook:~$ sh lab5.sh 5 5
10
(base) kirill@kirill-HP-ENVY-Notebook:~$
```

Рисунок 1.53 – Результат выполнения скрипта для задания 25

Вывод

В ходе выполнения лабораторной работы я изучил основные возможности языка программирования Shell с целью автоматизации процесса администрирования системы за счет написания и использования командных файлов.