

Отчет о прохождении 3 этапа внешних курсов

Продвинутые темы

Михайлова Полина Игоревна, НБИбд-03-24

Содержание

1	Цель работы	1
2	Задание	1
3	Теоретическое введение.....	1
4	Выполнение лабораторной работы.....	1
5	Сертификат.....	29
6	Выводы	29
	Список литературы	29

1 Цель работы

Ознакомиться с функционалом операционной системы Linux.

2 Задание



Просмотреть видео и на основе полученной информации пройти тестовые задания.



3 Теоретическое введение

Линукс - в части случаев GNU/Linux — семейство Unix-подобных операционных систем на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU, и, возможно, другие компоненты. Как и ядро Linux, системы на его основе, как правило, создаются и распространяются в соответствии с моделью разработки свободного и открытого программного обеспечения. Linux-системы распространяются в основном бесплатно в виде различных дистрибутивов — в форме, готовой для установки и удобной для сопровождения и обновлений, — и имеющих свой набор системных и прикладных компонентов, как свободных, так и проприетарных.

4 Выполнение лабораторной работы

3 Этап: (рис. 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41).



 2  ПМ 1

< Шаг 5 из 12 >

3.1 Текстовый редактор vim

Вы прошли больше 80% курса, оставьте отзыв

[Оставить отзыв](#) [Нет, спасибо](#)

Какую клавишу(и) нужно нажать на клавиатуре, чтобы выйти из редактора vim? Считайте, что вы только что открыли файл и вам сразу понадобилось выйти из редактора.

Верно решили **32 523** учащихся
Из всех попыток **69%** верных

Выберите один вариант из списка

☒ Хорошая работа.

☐ "q", затем "Enter"

☐ ":", затем "q"

☐ "Q"

☐ "Ctrl", затем "x"

☒ ":", затем "q", затем "Enter"

Следующий шаг

Решить снова

Figure 1: Задание 1

Стоит упомянуть, что у редактора vim есть tutorial, который позволяет разобраться с командами, необходимыми для стандартной работы. За выход из редактора отвечают следующие команды:


- ZQ - выйти без сохранения
- :q! - выйти без сохранения
- ZZ - записать файл и выйти (если файл не изменяли, то записываться он не будет)
- :wq - записать файл и выйти
- :x - записать файл и выйти
- :w - записать файл

- :sav filename - “сохранить как”
- :w filename - “сохранить как”
- :w! - записать файл

Как мы видим, вариантов много, при этом каждый сможет найти тот, который подойдёт под конкретную ситуацию.

Выберите все подходящие ответы из списка

Из всех попыток **20%** верных

 Верно. Так держать!

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☒ В этой строке 9 "слов" (word)
- ☐ В этой строке 9 "больших слов" (WORD)
- ☒ Нажимая только на W, нельзя переместить курсор на "."
- ☒ Чтобы попасть в конец строки, нужно совершить меньше нажатий на W, чем на w
- ☐ В этой строке 5 "слов" (word)
- ☐ Чтобы попасть в конец строки, нужно одинаковое число нажатий, что на W, что на w

Следующий шаг

Решить снова

Figure 2: Задание 2

Strange_ TEXT is_here. 2=2 YES!

Точка считается “маленьким словом”, так что всего их 9: Strange_, is_here, ., 2, =, 2, ! и два лишних пробела.

И если посчитать нажатия на w и на W, то действительно после 10 штук попадем в одно место. 10 нажатий на W, это то же самое, что и 10 нажатий на w,

В этих наборах нажатие на клавишу Esc обозначается как <Esc> (т.е. знаки < и > не несут отдельного смысла).

Примечание: во всех утверждениях имеется в виду, что мы находимся в редакторе vim, включен нормальный режим работы и курсор находится в самом начале строки.

Выберите все подходящие ответы из списка

Верно решили **23 655** учащихся
Из всех попыток **16%** верных

✓ Правильно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ d2wwywpp
- ☒ d2wwywPp
- ☐ x2wwywPp
- ☒ d2w\$bifour four <Esc>
- ☐ d2dwywPp
- ☒ xxxxxxxxwywPp

Следующий шаг

Решить снова

Figure 3: Задание 3

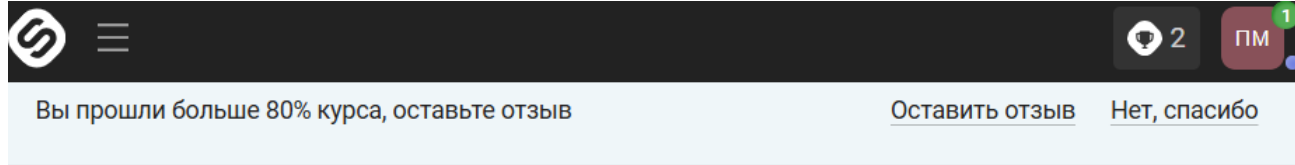
d2wwifour four <<Esc>>

d2wwywPp

d2w\$\$bifour four <<Esc>>

- \$ — в конец текущей строки;
- w — на слово вправо;
- b — на слово влево;
- i — начать ввод перед курсором;
- p — вставка содержимого неименованного буфера под курсором;
- P — вставка содержимого неименованного буфера перед курсором;
- yy (также Y) — копирование текущей строки в неименованный буфер;

- уу — копирование числа строк начиная с текущей в неименованный буфер;



Предположим, что вы открыли файл в редакторе vim и хотите заменить в этом файле все строки, содержащие слово `Windows`, на такие же строки, но со словом `Linux`. Если в какой-то строке слово `Windows` встречается больше, чем один раз, то заменить на `Linux` в этой строке нужно **только самое первое** из этих слов.

Какую команду нужно ввести для этого в vim? Укажите необходимую команду целиком (т.е. **включая** ввод ":" в самом начале), однако нажатие на `Enter` после ввода команды обозначать никак **не нужно**.

Напишите текст

✓ Так точно!

Верно решил **24 631** учащийся
Из всех попыток **57%** верных

```
:%s/Windows/Linux/
```

Figure 4: Задание 4

Поиск и замена в редакторе работают по следующей схеме:


`:{пределы}s/{что заменяем}/{на что заменяем}/{опции}`

Для замены во всем файле можно использовать символ %.

Выберите все подходящие ответы из списка

Верно решили **23 497** учащихся

Из всех попыток **29%** верных

 Правильно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☒ Режим выделения открывается из нормального режима по нажатию "v"
- ☒ Когда вы находитесь в режиме выделения, внизу редактора горит надпись – VISUAL -- (или – ВИЗУАЛЬНЫЙ РЕЖИМ --)
- ☒ Выйти из режима выделения можно, нажав клавишу Esc два раза
- ☐ Режим выделения открывается из любого другого режима по нажатию "v"
- ☐ Режим выделения открывается при помощи команды :visual
- ☐ Чтобы выйти из режима выделения, нужно ввести :q

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: **2 балла**

Figure 5: Задание 5

Команда \$ — в конец текущей строки, W - до пробела вправо - то есть, перемещение.

Нажать Esc достаточно один раз, но да ладно.

Надпись visual - горит.

d — используется совместно с командами перемещения. Удаляет символы с текущего положения курсора до положения после ввода команды перемещения.

yy (также Y) — копирование текущей строки в буфер;

Выберите один вариант из списка

Верно решили **30 266** учащихся
Из всех попыток **65%** верных

Правильно.

- ☒ Только из набора С
- ☐ Никакие команды появляться не будут
- ☐ Из наборов А и С
- ☐ Только из набора А
- ☐ Из наборов В и С

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: **1 балл**

Figure 6: Задание 6

Только из набора С потому что у каждой оболочки свой буфер, который при выходе из нее буде записываться в файл истории.



ПМ

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [script1.sh](#), [script2.sh](#).

Предположим, что вы находитесь в директории `/home/bi/Documents/` и запускаете в ней скрипт следующего содержания:

```
#!/bin/bash

cd /home/bi/
touch file1.txt
cd /home/bi/Desktop/
```

Как будет выглядеть **абсолютный путь** до созданного файла `file1.txt` по окончании работы скрипта?

Верно решили **29 905** учащихся
Из всех попыток **76%** верных

Выберите один вариант из списка



Абсолютно точно.

- ☐ `/home/bi/Desktop/file1.txt`
- ☒ `/home/bi/file1.txt`
- ☐ Никкак (файла `file1.txt` не будет существовать после завершения работы скрипта)
- ☐ `/home/bi/Documents/file1.txt`

Следующий шаг

Решить снова

Figure 7: Задание 7

`/home/bi/file1.txt` - потому что именно в этой директории мы создаем новый файл, а уже после его создания мы переходим в другую папку.

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте. [variables1.sh](#) и [variables2.sh](#).

Какие из представленных ниже строк **могут** быть именами переменных в bash? Выберите **все** подходящие варианты!

Подсказка: если все варианты ответов являются неверными, то не отмечайте ни один из них и нажимайте кнопку "Отправить"/"Submit".

Выберите все подходящие ответы из списка

Верно решили **27 188** учащихся
Из всех попыток **25%** верных

✓ Отлично!

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ var i able
- ☒ variable_123
- ☐ vari/able
- ☐ variab\$\$le
- ☐ var.i.able
- ☒ variable123
- ☐ var-i-able

Figure 8: Задание 8

Имя не может начинаться с цифры, содержать специальные символы или пробелы.

подсказка. В случае проблем с решением задачи, обратите внимание [на наши рекомендации по написанию скриптов](#).

**Напишите программу. Тестируется через stdin
→ stdout**

Верно решили **25 053** учащихся
Из всех попыток **41%** верных

✓ Так точно!

Теперь вам доступен [Форум решений](#), где вы можете сравнить свое решение с другими или спросить совета.

```
1 #!/bin/bash
2 echo "Arguments are: \"$1=$1 \"$2=$2"
3
4
5
6
7
```

Следующий шаг

Решить снова

Figure 9: Задание 9

\$ echo опции строка Эта команда печатает строки, которые передаются в качестве аргументов в стандартный вывод и обычно используется в сценариях оболочки для отображения сообщения или вывода результатов других команд.

var1=\$1 - обозначение переменных

var2=\$2

echo "Arguments are: \"\$1=\$var1 \"\$2=\$var2" - строка печати.

особенностями написания `$` в некоторых видах заданий на Stepik.

Выберите все подходящие ответы из списка

Верно решили **23 158** учащихся
Из всех попыток **16%** верных

☒ Хорошая работа.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☒ `-s $0`
- ☒ `! (4 -le 3)`
- ☐ `-n $1`
- ☒ `-z ""`
- ☐ `-z " "`
- ☐ `$# -gt 0`

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: **1 балл**

Figure 10: Задание 10

- `$0` - имя скрипта
- `$#` - вернет количество аргументов
- `-ge` - больше или равно
- `-n` - не пустая строка.

Имя скрипта - это не пустая строка.

`$#` Это число аргументов без учета имени скрипта, который всегда `$0`. И число аргументов всегда будет или равно нулю, или больше него, тк просто не может скатиться в отрицательную сторону.

Выберите один вариант из списка

Из всех попыток **64%** верных

✓ Правильно.

- ☐ Сначала four, потом one
- ☒ Сначала four, потом four
- ☐ Сначала one, потом two
- ☐ Сначала two, потом one

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: **1 балл**



Figure 11: Задание 11


- -lt, (<) - меньше
- -gt - больше
- -eq - равно


3 не больше 5, 3 не меньше 3, 3 не равно 4.

5 не больше 5, 5 не меньше 3, 5 не равно 4.

Оба раза выведет four.




 2

ПМ 

Напишите программу. Тестируется через stdin
→ stdout

Верно решили **23 310** учащихся
Из всех попыток **38%** верных

 Правильно.

Теперь вам доступен [Форум решений](#), где вы можете сравнить свое решение с другими или спросить совета.

```
1 #!/bin/bash
2
3 if [[ $1 -eq 0 ]]; then
4     echo "No students"
5 elif [[ $1 -eq 1 ]]; then
6     echo "1 student"
7 elif [[ $1 -eq 2 ]]; then
8     echo "2 students"
9 elif [[ $1 -eq 3 ]]; then
10    echo "3 students"
11 elif [[ $1 -eq 4 ]]; then
12    echo "4 students"
13 else
14    echo "A lot of students"
15 fi
```

Figure 12: Задание 12

1. Задаю общую часть в каждом выводе - слово "student": v=student
2. Выполняем команды для разных аргументов.
3. res - это результат для вывода
4. echo "\$res" - вывести результат

Верно решили 24 002 учащихся

Из всех попыток 45% верных

Выберите один вариант из списка

☒ Верно. Так держать!

- ☐ 5 раз "start" и ни разу "finish"
- ☒ 5 раз "start" и 4 раза "finish"
- ☐ 5 раз "start" и 5 раз "finish"
- ☐ 5 раз "start" и 2 раза "finish"

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: **1 балл**

Figure 13: Задание 13

- (Start)
- $a > c$ нет (Finish)
- (Start)
- $, > c$ нет (Finish)
- (Start)
- $b > c$ нет (Finish)
- (Start)
- $, > c$ нет (Finish)
- (Start)
- $c_d > c$ да

```
./script.sh
enter your name:
Egor
enter your age:
16
Egor, your group is child
enter your name:
Elena
enter your age:
0
bye
```

№2:

```
./script.sh
enter your name:
Elena Petrovna
enter your age:
25
Elena Petrovna, your group is youth
enter your name:

bye
```

Подсказка: в случае проблем с решением задачи, обратите внимание [на наши рекомендации по написанию скриптов](#).

Напишите программу. Тестируется через stdin
→ stdout

Верно решили **21 670** учащихся
Из всех попыток **23%** верных


-  Прекрасный ответ.

Figure 14: Задание 14

```

1 # put your shell (bash) code here
2 #!/bin/bash
3
4 while true
5 do
6     echo "enter your name:"
7     read name
8     if [[ -z "$name" ]]; then
9         echo "bye"
10        break
11    fi
12
13    echo "enter your age:"
14    read age
15    if [[ "$age" -eq 0 ]]; then
16        echo "bye"
17        break
18    fi
19
20    if [[ "$age" -le 16 ]]; then
21        group="child"
22    elif [[ "$age" -le 25 ]]; then
23        group="youth"
24    else
25        group="adult"
26    fi
27
28    echo "$name, your group is $group"
29 done
30

```

Figure 15: Задание 14

```

child=16
adult=25
stdout=0

```

```

while [[ $stdout != 1 ]] #конструкция типа while-True
do
    echo "enter your name: " #Пользователь вводит имя
    read name
    if [[ (-z $name) || ($name = 0) ]] ;then #Если имя не по параметрам, простим
ся
        echo "bye"
        stdout=1
    elif [[ -n $name ]]; then #А вот если имя нормальное
        while [[ $stdout != 1 ]] ;do
            echo "enter your age: " #То пусть вводит возраст

```



```

        read age #Считываем возраст
        if [[ ($age -eq 0) || (-z $age) ]] ;then #Если возраст 0 или строка
пуста - прощаемся
            echo "bye"
            stdout=1
        elif [[ $age -le $child ]] ;then #Если меньше или равен ребенку, то
ребенок
            echo "$name, your group is child"
        elif [[ $age -gt $adult ]] ; then #Больше взрослого - то взрослый
            echo "$name, your group is adult" ;else
        if [[ ($age -ge 17) && ($age -le 25) ]] ;then #Если от 17 до 25,
то подросток.
            echo "$name, your group is youth" ;fi
        fi ;break
    done ;fi
done

```

Выберите все подходящие ответы из списка

Из всех попыток 20% верных

☒ Здорово, всё верно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ a+=\$b
- ☒ let "a=\$a+\$b"
- ☒ let "a = a + b"
- ☒ let "a+=b"
- ☒ let a=a+b

Следующий шаг

Решить снова

Ваши решения Вы получили: 1 балл

Figure 16: Задание 15

1. a = \$a
2. a += b это то же самое, что и a = a + b, но с символами "+" != "="
3. если выражение не в скобках, но с пробелами - работать не будет. (let a=a+b - сработает; let a = a + b - нет)

Верно решили 23 077 учащихся
Из всех попыток 51% верных

Выберите один вариант из списка

☒ Так точно!

- ☐ pwd
- ☐ /home/bi/Documents
- ☒ /home/bi
- ☐ `pwd`
- ☐ Код возврата команды pwd (0 в случае успешного выполнения и не 0 в случае ошибок)

[Следующий шаг](#) [Решить снова](#)

[Ваши решения](#) Вы получили: 1 балл

4.

Figure 17: Задание 16

Выведет путь до директории, в которую мы перешли, так как “pwd” - это команда

Выберите все подходящие ответы из списка

Из всех попыток 20% верных

☒ Правильно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ Сначала var=`program`, затем if [[\$var -eq 0]]
- ☐ if [[`program` -eq 0]]
- ☒ Сначала запустить program, затем if [[\$? -eq 0]]
- ☐ Ничего сделать нельзя
- ☒ if `program` > some_file.txt`

[Следующий шаг](#) [Решить снова](#)

[Ваши решения](#) Вы получили: 1 балл

Figure 18: Задание 16_2

programm выполняет стандартный вывод в терминал (если это принцип работы программы). И нам нужно настроить вывод в файл.

подсказка. Этот пример можно решить в уме, но если система проверки не принимает ваше решение, то возможно вы что-то упустили (возможно что-то совсем небольшое/невидимое 😊). В этом случае имеет смысл написать небольшой скрипт на bash, который проделает ровно то, что указано в задании и посимвольно сверить свой ответ с тем, что он выдаст на экран.

Верно решили **20 009** учащихся
Из всех попыток **28%** верных

Напишите текст

✓ Верно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

counters are and 110

Следующий шаг Решить снова

[Ваши решения](#) Вы получили: **2 балла**

Figure 19: Задание 17

Первая переменная локальная, и это просто пустая строка, вторая переменная - это сумма арифметической прогрессии от 1 до 10, равна 55, но при умножении на 2 даст 110.

второго **N**. Если же наоборот, **M** меньше **N**, то запускаем функцию `gcd` с первым аргументом **M**, а вторым **(N-M)**.

Пример корректной работы скрипта:

```
./script.sh
10 15
GCD is 5
7 3
GCD is 1
bye
```

Примечание: в вызове функции из себя самой нет ничего страшного или неправильного, т.ч. смело вызывайте `gcd` прямо внутри `gcd` !

Примечание 2: для завершения работы функции в произвольном месте, можно использовать инструкцию `return` (все инструкции функции после `return` выполняться не будут). В отличии от `exit` эта команда завершит только функцию, а не выполнение всего скрипта целиком. Однако в данной задаче можно обойтись и без использования `return` !

Figure 20: Задание 18

```
1 # put your shell (bash) code here
2 #!/bin/bash
3
4 gcd() {
5     if [ "$1" -eq "$2" ]; then
6         echo "GCD is $1"
7     elif [ "$1" -gt "$2" ]; then
8         gcd "$(($1 - $2))" "$2"
9     else
10        gcd "$1" "$(($2 - $1))"
11    fi
12 }
13
14 while true; do
15     read a b
16     if [ -z "$a" ]; then
17         echo "bye"
18         break
19     fi
20     gcd "$a" "$b"
21 done
22
23
24
25
```

Figure 21: Задание 18

Алгоритм нахождения НОД делением

1. Большее число делим на меньшее.
2. Если делится без остатка, то меньшее число и есть НОД (следует выйти из цикла).
3. Если есть остаток, то большее число заменяем на остаток от деления.
4. Переходим к пункту 1.

Чтобы проверить работу скрипта, вы можете записать сразу несколько команд в файл и передать его скрипту на stdin (т.е. выполнить `./script.sh < input.txt`). В этом случае он должен вывести сразу все ответы на экран.

Например, если входной файл будет следующего содержания:

```
10 + 1
2 ** 10
exit
```

то на экране будет:

```
11
1024
bye
```

Если же на вход поступит следующий файл:

```
3 - 5
2/10
exit
```

то на экране будет:

```
2
ergor
```

т.к. вторая команда была **некорректной** (в ней всего один аргумент, т.к. нет пробелов между числами и операцией, а единственная допустимая команда из одного аргумента это "exit").

5.

Figure 22: Задание 1

```

1 while true
2 do
3     read num1 op num2
4     if [[ $num1 == "exit" ]]
5     then
6         echo "bye"
7         exit
8     elif [[ $num1 -ne ^[0-9]+$ || $num2 -ne ^[0-9]+$ ]]
9     then
10        echo "error"
11        exit
12    else
13        case $op in
14            "+") let "res = num1 + num2";;
15            "-" ) let "res = num1 - num2";;
16            "/" ) let "res = num1 / num2";;
17            "*" ) let "res = num1 * num2";;
18            "%" ) let "res = num1 % num2";;
19            "**" ) let "res = num1 ** num2";;
20            *) echo "error" ; exit ;;
21        esac
22        echo "$res"
23    fi
24 done

```

Figure 23: Задание 19

Калькулятор выглядит обычно - мы вводим два числа, пишем, что с ними надо сделать, и потом, учитывая случаи ошибок, выводим результат.

Верно решили **20 547** учащихся
Из всех попыток **36%** верных

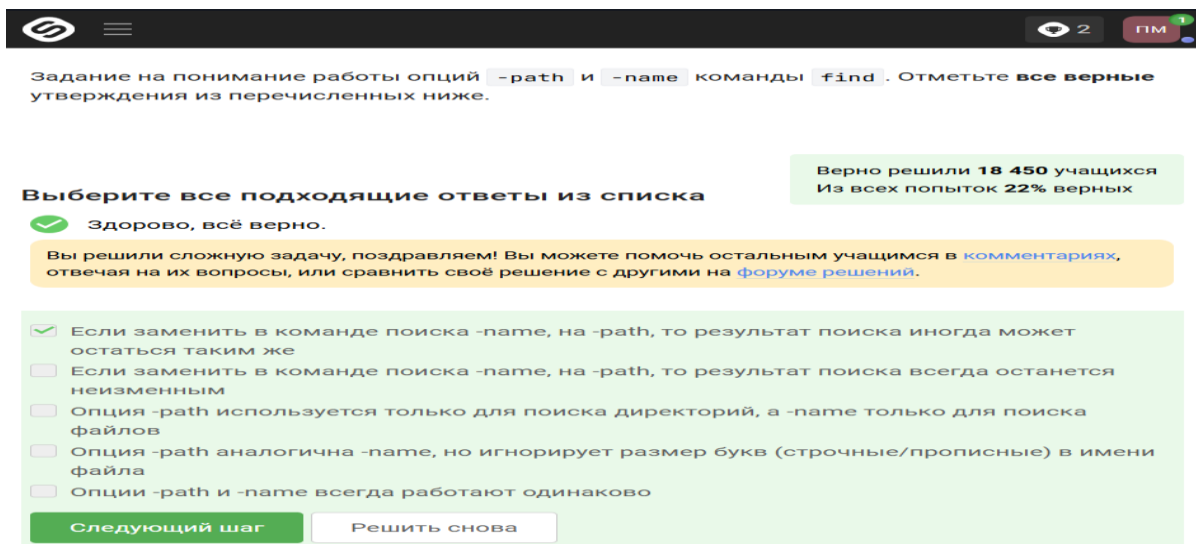
Выберите все подходящие ответы из списка

☒ Абсолютно точно.

- ☒ STARS.txt
- ☒ Star_Wars.avi
- ☐ Eddard_Stark_biography.txt
- ☐ star_trek_OST.mp3
- ☐ stardust.mpeg

Figure 24: Задание 20

-iname ищет без учета регистра, а -name в точности как в запросе. Звездочка стоит после слова - это значит после слова может быть сколько угодно символов.



Задание на понимание работы опций `-path` и `-name` команды `find`. Отметьте **все верные** утверждения из перечисленных ниже.

Выберите все подходящие ответы из списка

Верно решили **18 450** учащихся
Из всех попыток **22%** верных

✓ Здорово, всё верно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☒ Если заменить в команде поиска `-name`, на `-path`, то результат поиска иногда может остаться таким же
- ☐ Если заменить в команде поиска `-name`, на `-path`, то результат поиска всегда останется неизменным
- ☐ Опция `-path` используется только для поиска директорий, а `-name` только для поиска файлов
- ☐ Опция `-path` аналогична `-name`, но игнорирует размер букв (строчные/прописные) в имени файла
- ☐ Опции `-path` и `-name` всегда работают одинаково

Следующий шаг Решить снова

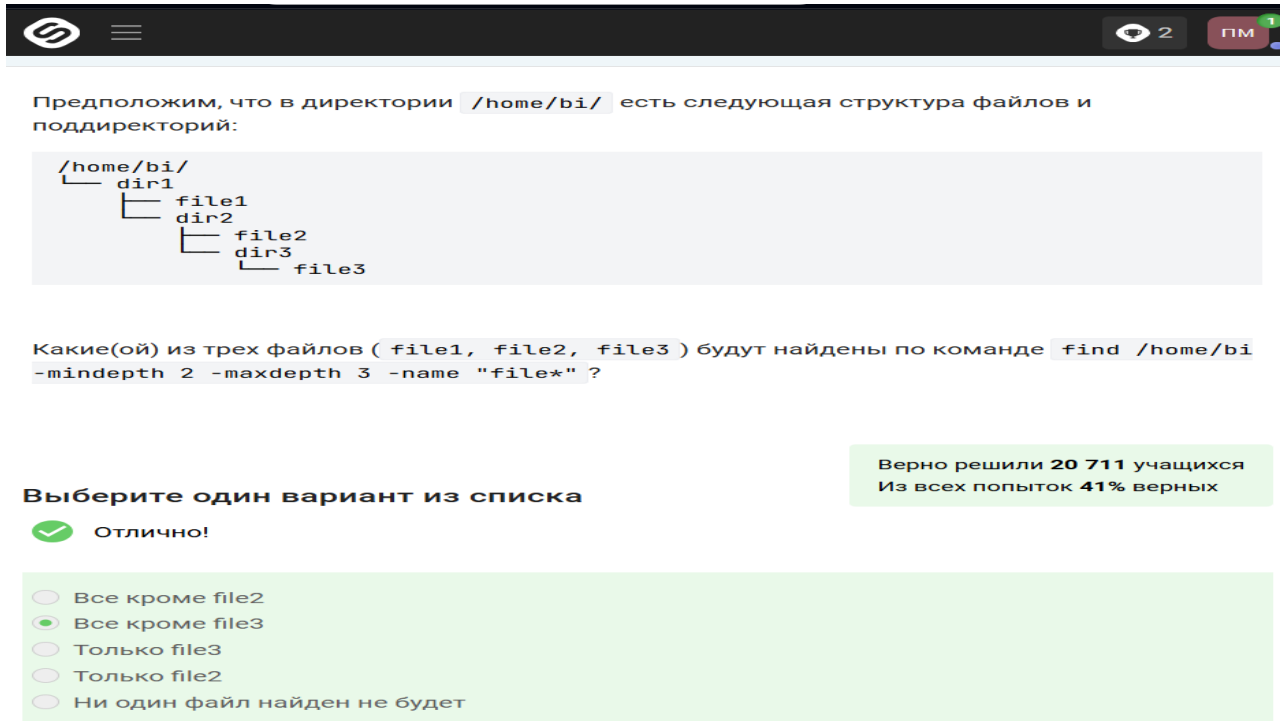
Figure 25: Задание 21

`find [path] [expression]`

где: `path` - это путь к директории, в которой нужно выполнить поиск файлов (по умолчанию, поиск производится в текущей директории и всех ее поддиректориях);

`expression` - это выражение, которое определяет критерии поиска файлов.

`-name`: поиск файлов по имени. Например: `find /home/user -name myfile.txt`



Предположим, что в директории `/home/bi/` есть следующая структура файлов и поддиректорий:

```

/home/bi/
├── dir1
│   ├── file1
│   └── dir2
│       ├── file2
│       └── dir3
│           └── file3

```

Какие(ой) из трех файлов (`file1`, `file2`, `file3`) будут найдены по команде `find /home/bi -mindepth 2 -maxdepth 3 -name "file*"` ?

Выберите один вариант из списка

Верно решили **20 711** учащихся
Из всех попыток **41%** верных

✓ Отлично!

- ☐ Все кроме `file2`
- ☒ Все кроме `file3`
- ☐ Только `file3`
- ☐ Только `file2`
- ☐ Ни один файл найден не будет

Figure 26: Задание 22

Текущий каталог - это `depth=1`, а остальное считается просто:

/home/bi -> depth=1

/home/bi/dir1 -> depth=2

/home/bi/dir1/dir2 -> depth=3

Задание на понимание работы опций `-A`, `-B` и `-C` команды `grep`. Пусть у вас есть файл `file.txt` из 10 строк, причем **в каждой строке есть** слово `"word"`. Если вы выполните на этом файле команды:

```
grep "word" file.txt > results.txt
grep -A 1 "word" file.txt > results.txt
grep -B 1 "word" file.txt > results.txt
grep -C 1 "word" file.txt > results.txt
```

то какая(ие) из них создаст файл `results.txt` наибольшего размера?

Выберите один вариант из списка

Верно решили 20 237 учащихся
Из всех попыток 41% верных

✓ Правильно.

- ☒ results.txt будет одинакового размера во всех случаях
- ☐ Все, кроме `grep "word" file.txt > results.txt`
- ☐ `grep -C 1 "word" file.txt > results.txt`
- ☐ `grep -A 1 "word" file.txt > results.txt`
- ☐ `grep -A 1 "word" file.txt > results.txt` и `grep -B 1 "word" file.txt > results.txt`

Следующий шаг

Решить снова

Figure 27: Задание 23

Из описания `man`: Print NUM lines of trailing context after/before matching lines “matching lines” - множественное число, строки в которых нашлось совпадение

Т.е. если идут 2...10...100 строк подряд, в которых обнаружилось совпадение, контекст будет выведен до и после этой ГРУППЫ строк, а не до и после каждой строки в этой группе

Верно решили 18 706 учащихся
Из всех попыток 23% верных

Выберите все подходящие ответы из списка

✓ Всё правильно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ Well, xubuntu is OK
- ☐ Uuuubuntu!
- ☐ Lubuntu is better than Windows
- ☐ Mac OS X 10.9, Windows XP, Ubuntu 12.04
- ☐ Kbuntu
- ☒ Lubuntu is better than Ubuntu

Следующий шаг Решить снова

Ваши решения Вы получили: 2 балла

Figure 28: Задание 24

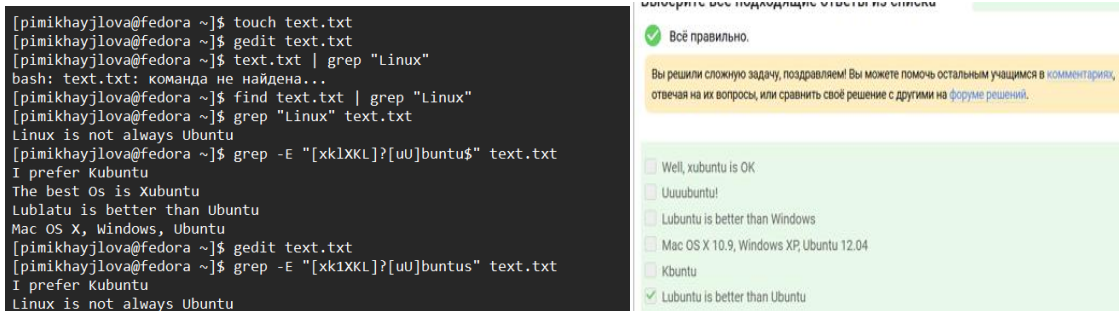


Figure 29: Задание 24

Объяснение на втором скриншоте.

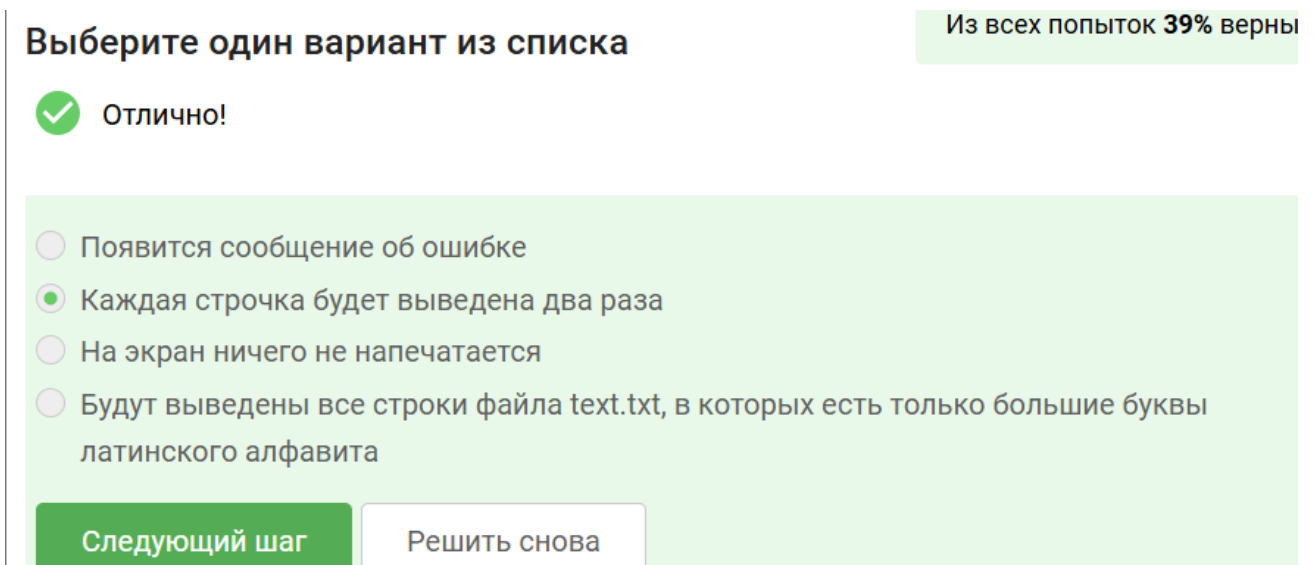


Figure 30: Задание 25

The -n option disables the automatic printing, which means the lines you don't specifically tell it to print do not get printed, and lines you do explicitly tell it to print (e.g. with p) get printed only once.

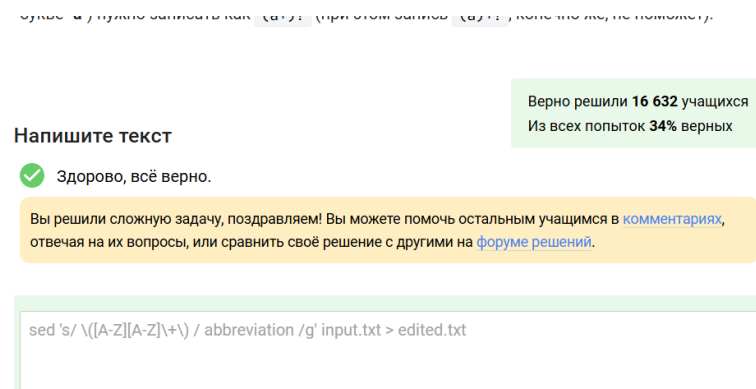


Figure 31: Задание 26

аббревиатура ABBA отличается от двух других аббревиатур тем, что справа он неё стоит запятая без пробела: “ABBA,”.

При этом по условию аббревиатура должна выглядеть как [XX] или [XXX] (и ещё больше X). Следовательно, для этой проверки надо добавить пробел квадратными скобками [] слева и, соответственно, с права.

Выберите один вариант из списка

Из всех попыток **51%** верных

☒ Верно.

☐ Графики и так не закрываются автоматически при закрытии gnuplot!

☐ Такой опции не существует

☐ -raise

☒ -p, --persist

Следующий шаг

Решить снова

Figure 32: Задание 27

-persist lets plot windows survive after main gnuplot program exits.

set key autotitle columnhead
plot 'data.csv' using 1:2

Какое в этом случае будет **название** у построенного **ряда данных** и **сколько** будет нарисовано **точек** на графике?

Выберите один вариант из списка

Верно решили **17 975** учащихся
Из всех попыток **32%** верных

☒ Всё правильно.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ Название "data.csv" using 1:2, нарисовано 10 точек
- ☐ Название – первое значение из первого столбца, нарисовано 10 точек
- ☐ Название – первое значение из первого столбца, нарисовано 9 точек (точка из первой строки пропущена)
- ☐ Название – первое значение из второго столбца, нарисовано 10 точек
- ☒ Название – первое значение из второго столбца, нарисовано 9 точек (точка из первой строки пропущена)

Figure 33: Задание 28

plot 'data.csv' using 1:2 даст ошибку:

warning: Skipping data file with no valid points ^ x range is invalid

Скорее всего причиной такого поведения является тот факт, что формат CSV содержит строки, где столбцы разделены запятой? Содержимое файла:

1,21
2,22
3,23
4,24
5,25
6,26
7,27
8,28
9,29
10,30

Верно решили 13 935 учащихся
Из всех попыток 44% верных

Напишите текст

✓ Здорово, всё верно.

```
set xtics ("point 1, value ".x1 x1, "point 2, value ".x2 x2, "point 3, value ".x3 x3)
```

Следующий шаг

Решить снова

Figure 34: Задание 29

Сначала идет команда установки подписей, а потом в скобках:

подпись - пробел - переменная с координатой - запятая

Повторяется это количество раз соответствующее числу переменных, и без запятой (в случае с последней переменной)

А подпись в свою очередь получается конкатенацией текста из задания и переменной с координатой.

Верно решили 12 854 учащихся
Из всех попыток 47% верных

Напишите текст

✓ Верно. Так держать!

```
a=a+1  
zrot=(zrot+350)%360  
set view xrot,zrot  
splot -x**2-y**2  
pause 0.1  
if (a<50) reread
```

Следующий шаг

Решить снова

[Ваши решения](#) Вы получили: 3 балла

Figure 35: Задание 30

1. График строится строкой "splot x2+y2".
2. Вращение задается строкой "zrot=(zrot+10)%360". Значит, смещение вперед (которое было изначально) можно также задать строкой

“ $zrot=(zrot+360+10)\%360$ ” или иначе говоря “ $zrot=(zrot+370)\%360$ ”. А теперь посмотрим на наше требование - чтоб вращалось в другую сторону, значит, по аналогии, необходимо вместо перебора на 10 сделать недобор.

“ $zrot=(zrot+350)\%360$ ”

3. Строка “pause 0.2” ставит выполнение на паузу на определенный промежуток времени. В задании сказали перерисовывать чаще, значит пауза должна быть меньше.

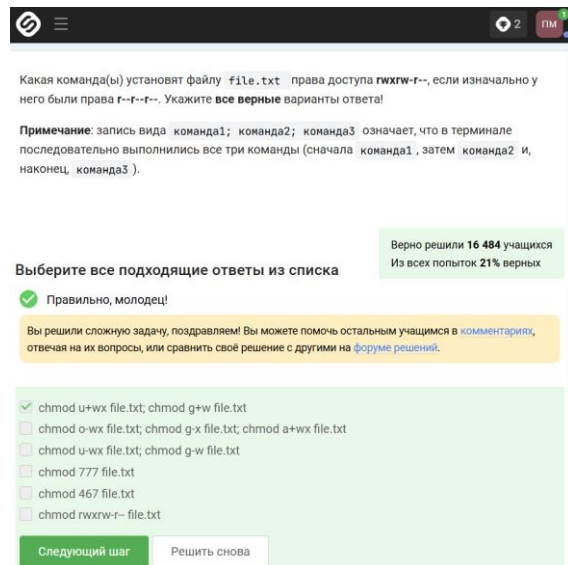


Figure 36: Задание 31

- r - чтение;
- w - запись;
- x - выполнение;
- s - выполнение от имени суперпользователя (дополнительный);
- u - владелец файла;
- g - группа файла;
- o - все остальные пользователи;
- 0 - никаких прав;
- 1 - только выполнение;
- 2 - только запись;
- 3 - выполнение и запись;
- 4 - только чтение;
- 5 - чтение и выполнение;

- 6 - чтение и запись;
- 7 - чтение запись и выполнение.

Выберите все подходящие ответы из списка Из всех попыток 15% верных

✓ Правильно, молодец!

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☐ chmod o+w dir
- ☐ sudo chmod g+w dir
- ☐ sudo chown :group dir
- ☐ chown user:group dir
- ☒ sudo chmod o+w dir
- ☒ sudo chown user:group dir

[Следующий шаг](#) [Решить снова](#)

[Ваши решения](#) Вы получили: 1 балл

Figure 37: Задание 32

Решений два типа:

- Сменить права гостей, добавив W
- Сделать владельцем нужную группу или пользователя, в зависимости от того, у кого из них уже есть права на W
- Помнить, что root - владелец и остальные для него - others.

Выберите все подходящие ответы из списка Из всех попыток 21% верных

✓ Прекрасный ответ.

Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#).

- ☒ Размер файла в байтах
- ☐ Количество определенных букв (например, количество букв "А")
- ☒ Количество строк
- ☒ Количество слов
- ☐ Количество предложений

[Следующий шаг](#) [Решить снова](#)

[Ваши решения](#) Вы получили: 1 балл

Figure 38: Задание 33

- `wc -l` вывести количество строк
- `wc -c` вывести количество байт
- `wc -m` вывести количество символов
- `wc -L` вывести длину самой длинной строки
- `wc -w` вывести количество слов

Напишите текст

Верно решил 16 381 учащийся
Из всех попыток 53% верных

✓ Правильно.

```
du -sh .
```

Следующий шаг Решить снова

Ваши решения Вы получили: 2 балла

Figure 39: Задание 34

-h, –human-readable print sizes in human readable format (e.g., 1K 234M 2G)

-s, –summarize display only a total for each argument

Напишите текст

Из всех попыток 40% верных

✓ Хорошая работа.

```
mkdir dir{1..3}
```

Следующий шаг Решить снова

Figure 40: Задание 35

Команда создаст три директории от dir1 до dir3.

5 Сертификат

Figure 41: Сертификат

Figure 41: Сертификат

6 Выводы

Я просмотрела курс и освежила в памяти навыки работы с более сложными командами в Линукс.

Список литературы

1. Введение в Linux