

Внешний курс по Linux - Раздел 3

Отчёт по выполнению курса

Чистов Даниил Максимович

Содержание

1	Цель работы	4
2	Выполнение внешнего курса	5
3	Выводы	16

Список иллюстраций

2.1	Задание 1	5
2.2	Задание 2	6
2.3	Задание 3	6
2.4	Задание 4	7
2.5	Задание 5	7
2.6	Задание 6	8
2.7	Задание 7	8
2.8	Задание 8	8
2.9	Задание 9	9
2.10	Задание 10	9
2.11	Задание 11	10
2.12	Задание 12	10
2.13	Задание 13	11
2.14	Задание 14	11
2.15	Задание 15	12
2.16	Задание 16	12
2.17	Задание 17	13
2.18	Задание 18	13
2.19	Задание 19	14
2.20	Задание 20	15

1 Цель работы

Цель данной работы - пройти внешний курс по ОС Linux, чтобы проверить свои знания, а также узнать что-то новое. Выполнение требуется задокументировать.

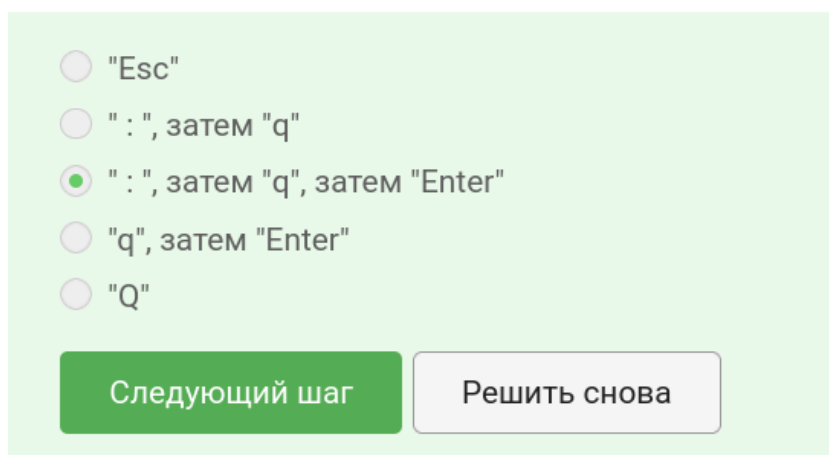
2 Выполнение внешнего курса

Все скриншоты первого этапа: (рис. [2.1]), (рис. [2.2]), (рис. [2.3]), (рис. [2.4]), (рис. [2.5]), (рис. [2.6]), (рис. [2.7]), (рис. [2.8]), (рис. [2.9]), (рис. [2.10]), (рис. [2.11]), (рис. [2.12]), (рис. [2.13]), (рис. [2.14]), (рис. [2.15]), (рис. [2.16]), (рис. [2.17]), (рис. [2.18]), (рис. [2.19]), (рис. [2.20]).

Стандартная процедура выхода из vim

Выберите один вариант из списка

☒ Здорово, всё верно.



- ☐ "Esc"
- ☐ " : ", затем "q"
- ☒ " : ", затем "q", затем "Enter"
- ☐ "q", затем "Enter"
- ☐ "Q"

Следующий шаг Решить снова

Рис. 2.1: Задание 1

Чтобы выполнить данное задание пришлось лично перепроверить каждый вариант. Это можно увидеть на видео.

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 5 "слов" (word)
- ☒ После 10 нажатий на W курсор окажется там же, где бы он был после 10 нажатий на w
- ☒ Чтобы попасть в конец строки, нужно совершить меньше нажатий на W, чем на w
- ☐ В этой строке 12 "слов" (word)
- ☒ В этой строке 5 "больших слов" (WORD)
- ☐ Чтобы попасть в конец строки, нужно одинаковое число нажатий, что на W, что на w

[Следующий шаг](#) [Решить снова](#)

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

Рис. 2.2: Задание 2

Аналогично с прошлым заданием, также пришлось открыть vi и проверить.

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

- ☒ ddithree four four four five<Esc>
- ☒ d2wwifour four <Esc>
- ☒ d2w\$bifour four <Esc>
- ☒ d2wwywPp
- ☒ xxxxxxxxwywPp
- ☐ d2wwywpp

[Следующий шаг](#) [Решить снова](#)

Рис. 2.3: Задание 3

Существует определённая схема действий по поиску и замене.

нажать на `Esc` после ввода команды обязательно никак не влияет.

Напишите текст

✓ Верно.

:%s/Windows/Linux

Следующий шаг

Решить снова

Рис. 2.4: Задание 4

Лично пришлось проверить.

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"
- ☐ Режим выделения открывается из любого другого режима по нажатию "v"
- ☐ Режим выделения открывается при помощи команды :visual
- ☒ В режиме выделения можно использовать команды d (удалить) и у (скопировать)
- ☒ Выйти из режима выделения можно, нажав клавишу Esc два раза
- ☒ В режиме выделения можно использовать команды перемещения (например, W, e, \$, и др.)

Следующий шаг

Решить снова

Рис. 2.5: Задание 5

Только из набора C.

Выберите один вариант из списка

✓ Прекрасный ответ.

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

Следующий шаг Решить снова

Рис. 2.6: Задание 6

В этой директории мы создаём файл, поэтому выбираем такой ответ.

✓ Прекрасный ответ.

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

Следующий шаг Решить снова

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

Рис. 2.7: Задание 7

Догадался о других вариантах, зная, что `_variable` и `variable` правильные

var i able

variab\$\$le

✓ `_variable`

✓ `_variable`

✓ `variable`

var.i.able

✓ `variable123`

Следующий шаг Решить снова

Рис. 2.8: Задание 8

Простая программа, вспомнил второй семестр

```
1 #!/bin/bash
2
3 var_1=$1
4 var_2=$2
5
6 echo "Arguments are: \$1=$var_1 \$2=$var_2"
7
8
9
10
```

Следующий шаг Решить снова

Ваши решения Вы получили: ••• из 3

Рис. 2.9: Задание 9

5 больше или равно 5 - верно. 4 не меньше или равно 3 - верно.

Примечание: если вы планируете проверять варианты ответов у себя в терминале, обратите внимание на то, что содержащие символ `$` тексты *могут* изменяться при копировании — не забудьте отредактировать их в соответствии с изображением на экране. Это связано с особенностями написания `$` в некоторых видах заданий на Stepik.

Выберите все подходящие ответы из списка

✓ Отличное решение!

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](#).

Верно решили 23 158 уч
Из всех попыток 16% ве

- ☐ -z " "
- ☒ 5 -ge 5
- ☐ \$# -gt 0
- ☐ \$var1 == \$var2 && \$var1 != \$var2
- ☐ -n \$1
- ☒ ! (4 -le 3)

Следующий шаг Решить снова

Рис. 2.10: Задание 10

При первом запуске с `var=3` условие `elif "[[$var -lt 3]]"` будет ложным, выполнится `else`. При втором запуске с `var=5` также выполнится последний `else` блок.

Выберите один вариант из списка

✓ Хорошая работа.

Верно решили 25 138 учащихся
Из всех попыток 64% верных

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

Следующий шаг
 Решить снова

[Ваши решения](#)
 Вы получили: 1 из 1

Рис. 2.11: Задание 11

Здесь case'ом проверяем аргумент командной строки \$1, и выводим в зависимости от этого аргумента: “No students” для значения 0, количество студентов в виде “\$1 student(s)” для 1 или от 2 до 4, и “A lot of students” для всех остальных значений.

```

1 #!/bin/bash
2
3 s=student
4 case $1 in
5 0) res="No ${s}s";;
6 1) res="${s} ${s}";;
7 [2-4]) res="${s} ${s}s";;
8 *) res="A lot of students";;
9 esac
10
11 echo "$res"
12
13
14
15
16
  
```

Следующий шаг
 Решить снова

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

Рис. 2.12: Задание 12

Цикл for проходит по 5 элементам, и echo “start” выполняется 5 раз. Команда echo “finish” пропускается один раз из-за условия if [[\$str > “c”]], поэтому выполняется 4 раза.

Выберите один вариант из списка

✓ Хорошая работа.

Верно решили 24 582 учащихся
Из всех попыток 45% верных

☐ 3 раза "start" и 2 раза "finish"
☒ 5 раз "start" и 4 раза "finish"
☐ 3 раза "start" и 3 раза "finish"
☐ 3 раза "start" и ни разу "finish"

Следующий шаг
 Решить снова

Ваши решения Вы получили: 1 балл из 1

1206 169 Шаг 8
 Следующий шаг >

Рис. 2.13: Задание 13

Прописываю условия для “bye”, спрашиваю имя, спрашиваю возраст и elif’ами определяю человека в группу и вывожу соответствующее сообщение.

```

1 child=16
2 adult=25
3 stdout=0
4
5 while [[ $stdout != 1 ]]
6 do
7     echo "enter your name: "
8     read name
9     if [[ (-z $name) || ($name = 0) ]]; then
10        echo "bye"
11        stdout=1
12    elif [[ -n $name ]]; then
13        while [[ $stdout != 1 ]]; do
14            echo "enter your age: "
15            read age
16            if [[ ($age -eq 0) || (-z $age) ]]; then
17                echo "bye"
18                stdout=1
19            elif [[ $age -le $child ]]; then
20                echo "$name, your group is child"
21            elif [[ $age -gt $adult ]]; then
22                echo "$name, your group is adult" ; else
23                if [[ ($age -ge 17) && ($age -le 25) ]]; then
24                    echo "$name, your group is youth" ; fi
25                fi ; break
26            done ; fi
27 done
  
```

Рис. 2.14: Задание 14

Эти варианты корректно увеличивают значение переменной a на значение переменной b в bash.

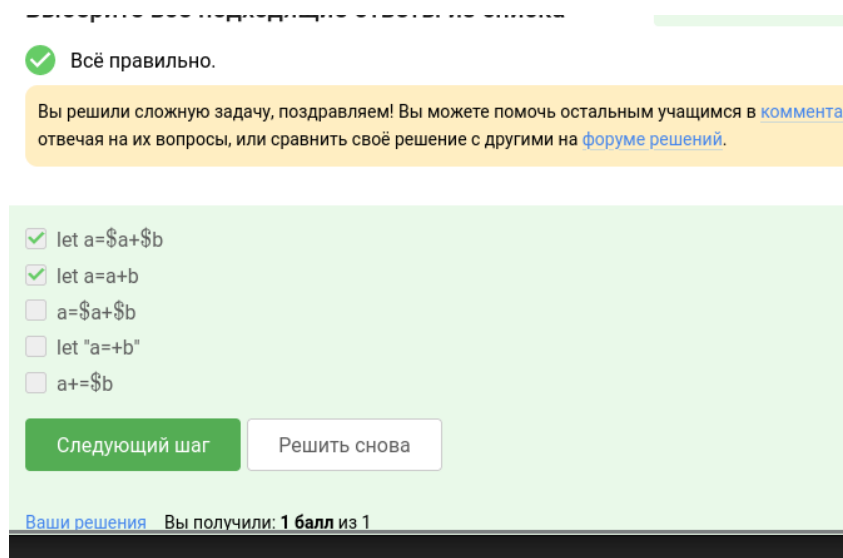


Рис. 2.15: Задание 15

В таком случае не напишется текст `pwd`, а будет выполнена команда `pwd`, а раз мы находимся в `/home/bi`, то и будет выведен такой ответ.

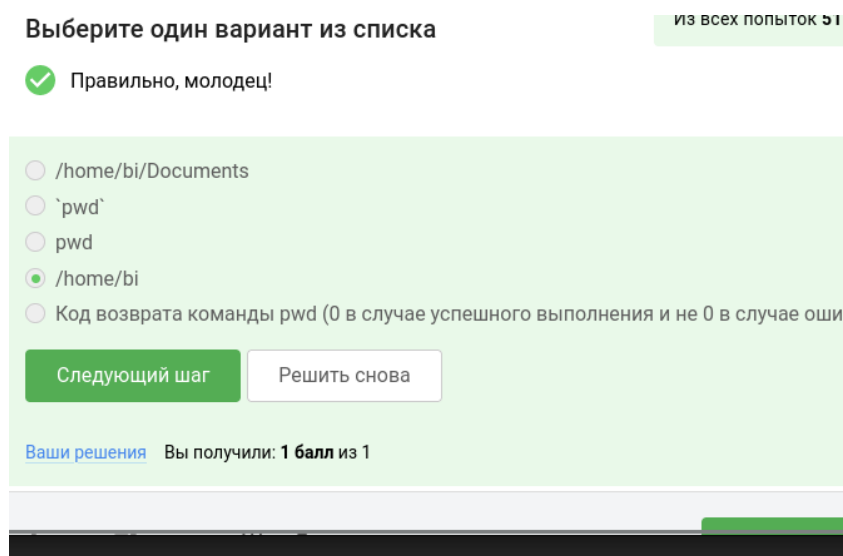


Рис. 2.16: Задание 16

“if program > some_file.txt” - Программа перенаправляется в `some_file.txt`, а if проверяет только код возврата программы (автоматически сохранён в `$?`).

“Сначала запустить program, затем `if [[$? -eq 0]]`” - запускается программа, и затем отдельно проверяется код возврата последней выполненной команды (код хранится в переменной \$?).

Выберите все подходящие ответы из списка

✓ Прекрасный ответ.

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

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

[Следующий шаг](#) [Решить снова](#)

[Ваше решение](#) [Ваше решение 1](#) [Ваше решение 1](#)

Рис. 2.17: Задание 17

После десяти вызовов функции с параметрами от 1 до 10 значения переменной \$c2 становятся 110 соответственно, \$c1 же не будет иметь никакого значения.

который проделает ровно то, что указано в задании и посимвольно сверить свой ответ

Напишите текст

✓ Здорово, всё верно.

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

counters are and 110

[Следующий шаг](#) [Решить снова](#)

Рис. 2.18: Задание 18

Программа запрашивает два числа, вычисляет их наибольший общий делитель (НОД) с помощью функции gcd (которая выполняет цикл с делением и нахождением остатка), выводит результат и повторяет процесс до ввода пустого значения, при котором завершает работу.

```
1 while [ true ]
2 do
3     read n1 n2
4     if [ -z $n1 ]; then
5         echo "bye"
6         break
7     else
8         gcd () {
9             remainder=1
10            if [ $n2 -eq 0 ]
11            then
12                echo "bye"
13            fi
14            while [ $remainder -ne 0 ]
15            do
16                remainder=$((n1%n2))
17                n1=$n2
18                n2=$remainder
19            done
20        }
21        gcd $1 $2
22        echo "GCD is $n1"
23    fi
24 done
```

Следующий шаг Решить снова

Ваши решения Вы получили: 4 балла из 4

Рис. 2.19: Задание 19

Команда `find /home/bi -mindepth 2 -maxdepth 3 -name "file*"` найдёт файлы на уровнях от 2 до 3 включительно, поэтому `file1` и `file2` будут найдены, а `file3` не будет найден.

```
└─ dir2
   └─ file2
      └─ dir3
         └─ file3
```

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

Выберите один вариант из списка

✔ Правильно.

- ☐ Только file3
- ☐ Только file1
- ☐ Все три файла
- ☒ Все кроме file3
- ☐ Все кроме file1

[Следующий шаг](#) [Решить снова](#)

[Ваши решения](#) Вы получили: ... из 1

Рис. 2.20: Задание 20

Мне хватило решить столько заданий, чтобы получить сертификат о прохождении

3 Выводы

По окончании выполнения раздела 2 внешнего курса по ОС Linux, я проверил свои знания, а также узнал что-то новое. Документация к выполнению была оформлена. Поставленные цели были достигнуты.