

Прохождение внешнего курса

Продвинутые темы

Комягин Андрей Николаевич

Содержание

1	Цель прохождения курса	5
2	Выполнение заданий курса	6
2.1	Общая информация о курсе	6
3	Вывод	23

Список иллюстраций

2.1	Вопрос №1	6
2.2	Вопрос №2	7
2.3	Вопрос №3	7
2.4	Вопрос №4	8
2.5	Вопрос №5	8
2.6	Вопрос №6	9
2.7	Вопрос №7	9
2.8	Вопрос №8	10
2.9	Вопрос №9	10
2.10	Вопрос №10	11
2.11	Вопрос №11	11
2.12	Вопрос №12	12
2.13	Вопрос №13	13
2.14	Вопрос №14	13
2.15	Вопрос №15	14
2.16	Вопрос №16	14
2.17	Вопрос №17	15
2.18	Вопрос №18	15
2.19	Вопрос №19	16
2.20	Вопрос №20	16
2.21	Вопрос №21	17
2.22	Вопрос №22	17
2.23	Вопрос №23	18
2.24	Вопрос №24	18
2.25	Вопрос №25	19
2.26	Вопрос №26	19
2.27	Вопрос №27	20
2.28	Вопрос №28	20
2.29	Вопрос №29	21
2.30	Вопрос №30	21
2.31	Вопрос №31	22
2.32	Вопрос №32	22

Список таблиц

1 Цель прохождения курса

Познакомиться с текстовым редактором `vim`, языком `bash`. Научиться писать скрипты. Освоить продвинутый поиск, постройку графиков. Углубиться в управление разрешениями

2 Выполнение заданий курса

2.1 Общая информация о курсе

Информация об этом дана в видеоролике(рис. 2.1).

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.

Выберите один вариант из списка

Верно решили 32 523 учащихся
Из всех попыток 69% верных

☒ Так точно!

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

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

👍 1146 🗨 443 Шаг 5 Следующий шаг >

Рис. 2.1: Вопрос №1

Решено путем практического эксперимента(рис. 2.2).

в то место справки, где описано это перемещение, а так как все перемещения описаны рядом, то двигаясь по тексту вверх и вниз можно прочитать и про `e` и про `b` и, самое главное, про `word` и `WORD`. Кроме того, можно вызвать сразу справку по термину `word` при помощи `:help word`. Чтобы закрыть справку, нужно ввести команду `:q`.

Выберите все подходящие ответы из списка

☒ Прекрасный ответ.

Верно решили **25 385** учащихся
Из всех попыток **20%** верных

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](#).

- ☐ Чтобы попасть в конец строки, нужно одинаковое число нажатий, что на `W`, что на `w`
- ☐ 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 9 "слов" (`word`)
- ☐ Нажимая только на `w`, нельзя переместить курсор на `"`.

Следующий шаг

Решить снова

Рис. 2.2: Вопрос №2

Решено путем практического эксперимента(рис. 2.3).

Предположим, что в текстовом файле записана одна единственная строка:

```
one two three four five
```

и вам нужно преобразовать её в строку

```
three four four four five
```

Какие(ой) из предложенных ниже **наборов нажатий клавиш** выполнят такое редактирование? 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

☒ Хорошая работа.

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

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](#).

- ☒ `xxxxxxxxxywPp`
- ☒ `d2w$bifour four <Esc>`
- ☐ `x2wwywwPp`
- ☒ `d2wwywwPp`
- ☒ `ddithree four four four five<Esc>`
- ☒ `d2wwifour four <Esc>`

Следующий шаг

Решить снова

Рис. 2.3: Вопрос №3

Используем `%s` для замены всех слов `Windows` на `Linux`(рис. 2.4).

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

✓ Хорошие новости, верно!

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

:%s/Windows/Linux

Следующий шаг

Решить снова

Ваши решения Вы получили: **2 балла** из 2

Рис. 2.4: Вопрос №4

Отвечил на вопрос, потому что попробовал данный режим(рис. 2.5).

Мы совсем не рассказали вам про третий режим работы vim – режим **выделения (Visual)**. Предлагаем вам ознакомиться с ним самостоятельно. Например, это можно сделать во время прохождения упражнений в vimtutor, который мы настоятельно рекомендуем вам для изучения vim!

Чтобы убедиться, что вы разобрались с этим режимом работы, отметьте, пожалуйста, **все верные** утверждения из списка ниже.

Подсказка: если вы не хотите проходить vimtutor целиком, то можете открыть его и поиском найти слово **"Visual"**. Вы попадете в задание, прохождение которого будет вполне достаточно, чтобы выполнить это задание.

Выберите все подходящие ответы из списка

✓ Верно.

Верно решили **23 497** учащихся
Из всех попыток **29%** верных

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](#).

- ✓ Когда вы находитесь в режиме выделения, внизу редактора горит надпись – VISUAL – (или – ВИЗУАЛЬНЫЙ РЕЖИМ –)
- ✓ В режиме выделения можно использовать команды `d` (удалить) и `y` (скопировать)
- ✓ Выйти из режима выделения можно, нажав клавишу `Esc` два раза
- ✓ В режиме выделения можно использовать команды перемещения (например, `W`, `e`, `$`, и др.)
- ☐ Режим выделения открывается при помощи команды `:visual`
- ✓ Режим выделения открывается из нормального режима по нажатию `"v"`

Следующий шаг

Решить снова

Рис. 2.5: Вопрос №5

Передвижение будет внутри последней оболочки(рис. 2.6).

Надеемся, что вы разобрались, что одну оболочку (например, `sh`) можно запустить из другой оболочки (например, из `bash`).

Предположим, что вы открыли терминал и у вас в нем запущена оболочка `bash`. Вы набираете в ней команды `A1`, `A2`, `A3`, а затем запускаете оболочку `sh`. В этой оболочке вы набираете команды `B1`, `B2`, `B3` и запускаете оболочку `bash`. И, наконец, в этой последней оболочке вы набираете команды `C1`, `C2`, `C3`. Если теперь вы попытаете при помощи стрелочек вверх/вниз перемещаться по истории набранных команд, то команды из какого набора(ов) будут появляться?

Выберите один вариант из списка

Верно решили 30 266 учащихся
Из всех попыток 65% верных

☒ Так точно!

☐ Только из набора A

☐ Никакие команды появляться не будут

☐ Из наборов A и C

☐ Из наборов B и C

☒ Только из набора C

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

Рис. 2.6: Вопрос №6

Переход в директорию первой командой `cd` и создание там файла. Последующий переход роли не играет(рис. 2.7).

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?

Выберите один вариант из списка

Верно решили 29 905 учащихся
Из всех попыток 76% верных

☒ Здорово, всё верно.

☐ Никак (файла `file1.txt` не будет существовать после завершения работы скрипта)

☒ `/home/bi/file1.txt`

☐ `/home/bi/Documents/file1.txt`

☐ `/home/bi/Desktop/file1.txt`

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

Рис. 2.7: Вопрос №7

Переменная не должна начинаться с цифры, не должна иметь символов кроме нижнего подчеркивания(рис. 2.8).

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

✓ 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](#).

- ☒ VARiable
- ☒ _variable
- ☐ var@iabile
- ☒ variable_123
- ☒ variable
- ☒ variable123
- ☒ __variable

Следующий шаг Решить снова

Рис. 2.8: Вопрос №8

Просто смотрим выражения и значения ключей. Можно проверить практической работой программы(рис. 2.9).

параметры или запустят ваш скрипт и какие в нем есть переменные.

Например, условие `0 -eq 0` **подходит**, т.к. ноль всегда равен нулю вне зависимости от аргументов и переменных внутри скрипта и на экран будет напечатано `True`. В то же время условие `$var1 -eq 0` **не подходит**, так как в переменной `var1` как может быть записан ноль (тогда будет напечатано `True`), так его может и не быть (тогда ничего напечатано не будет).

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

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

✓ Всё получилось!

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](#).

- ☒ \$# -ge 0
- ☒ ! (4 -le 3)
- ☒ \$var1 == \$var2 || \$var1 != \$var2
- ☒ 5 -ge 5
- ☒ -e \$0
- ☒ -s \$0

Следующий шаг Решить снова

Рис. 2.9: Вопрос №9

3 и 5 не подходят под условия “больше 5”, “меньше 3” или “равно 4”, поэтому попадают на ветку else(рис. 2.10).

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [branching2.sh](#), [branching3.sh](#).

Посмотрите на фрагмент bash-скрипта:

```
if [[ $var -gt 5 ]]
then
  echo "one"
elif [[ $var -lt 3 ]]
then
  echo "two"
elif [[ $var -eq 4 ]]
then
  echo "three"
else
  echo "four"
fi
```

Какие строки и в какой последовательности он выведет на экран, если сначала этот скрипт запустили задав переменную `var=3`, а затем запустили еще раз, но уже с `var=5`.

Выберите один вариант из списка

☒ Хорошие новости, верно!

Верно решили 25 138 учащихся
Из всех попыток 64% верных

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

Следующий шаг

Решить снова

Рис. 2.10: Вопрос №10

Простенькая программа с ветвлением. Для людей, умеющих программировать на других ЯП, проблем позникнуть не должно(рис. 2.11).

Примечание 6): в последней строке слово "lot" с маленькой буквы!

Примечание 2: в этой и всех последующих задачах на написание скриптов, если не указано явно, что нужно **проверять вход** (например, что он будет именно числом и именно от 0 до бесконечности), то этого делать **не нужно**!

Пример №1: если ваш скрипт называется `./script.sh`, то при запуске его как `./script.sh 1` на экране должно появиться:

```
1 student
```

Пример №2: если ваш скрипт называется `./script.sh`, то при запуске его как `./script.sh 5` на экране должно появиться:

```
A lot of students
```

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

Напишите программу. Тестируется через `stdin → stdout`

☒ Здорово, всё верно.

Верно решили 23 310 учащихся
Из всех попыток 38% верных

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

```
1 #!/bin/bash
2
3 v=$1
4
5 if [[ $v -eq 0 ]]
6 then
7   echo "No students"
8 elif [[ $v -eq 1 ]]
9 then
10  echo "$v student"
11 elif [[ $v -lt 5 ]]
12 then
13  echo "$v students"
14 else
15  echo "A lot of students"
16 fi
17
```

Рис. 2.11: Вопрос №11

Важное примечание: запятые также являются символами. Разделителем яв-

ляется только пробел(рис. 2.12).

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [loops1.sh](#), [loops2.sh](#).

Посмотрите на фрагмент bash-скрипта:

```
for str in a , b , c_d
do
  echo "start"
  if [[ $str > "c" ]]
  then
    continue
  fi
  echo "finish"
done
```

Если запустить этот скрипт, то **сколько раз** на экран будет выведено слово "start", а сколько раз слово "finish"?

Выберите один вариант из списка

☒ Прекрасный ответ.

Верно решили **24 582** учащихся
Из всех попыток **45%** верных

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

Следующий шаг

Решить снова

Рис. 2.12: Вопрос №12

Простейшее ветвление, обернутое в цикл while (рис. 2.13).

Напишите скрипт на `bash`, который будет определять в какую возрастную группу попадают пользователи. При запуске скрипт должен вывести сообщение `"enter your name:"` и ждать от пользователя ввода имени (используйте `read`, чтобы прочитать его). Когда имя введено, то скрипт должен написать `"enter your age:"` и ждать ввода возраста (опять нужен `read`). Когда возраст введен, скрипт пишет на экран `"Имя, your group is <группа>"`, где `<группа>` определяется на основе возраста по следующим правилам:

- младше либо равно 16: `"child"`,
- от 17 до 25 (включительно): `"youth"`,
- старше 25: `"adult"`.

После этого скрипт опять выводит сообщение `"enter your name:"` и всё начинается по новой (бесконечный цикл!). Если в какой-то момент работы скрипта будет введено пустое имя или возраст 0, то скрипт должен написать на экран `"bye"` и закончить свою работу (выход из цикла!).

Примеры корректной работы скрипта:

№1:

```
./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`

✓ Все получилось!

Верно решили 21 670 учащихся

Из всех попыток 23% верных

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

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

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

Рис. 2.13: Вопрос №13

Код целиком(рис. 2.14).

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

Рис. 2.14: Вопрос №14

Для обозначения арифметической операции используется `let`. Выражение должно быть обёрнуто кавычками или написано без пробелов(рис. 2.15).

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [math1.sh](#), [math2.sh](#).

Какие(ая) из предложенных ниже инструкций увеличат значение переменной `a` на значение переменной `b`? Например, если в `a` было записано 10, в `b` было 5, то в `a` должно записаться 15. Выберите **все подходящие** варианты!

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.

Подсказка: обратите особое внимание на кавычки и **пробелы**, они могут как принципиально изменить команду, так и ни на что не повлиять (в зависимости от команды и контекста)!

Выберите все подходящие ответы из списка

Верно решили 22 116 учащихся
Из всех попыток 20% верных

✓ Абсолютно точно.

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](#).

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

Рис. 2.15: Вопрос №15

Просто перемещаемся в директорию с помощью `cd` и выводим путь в текущую директорию(рис. 2.16).

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

```
#!/bin/bash
cd /home/bi/
echo "`pwd`"
```

Что в этом случае выведет команда `echo` на экран?

Выберите один вариант из списка

Верно решили 23 677 учащихся
Из всех попыток 51% верных

✓ Правильно, молодец!

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

Следующий шаг Решить снова

Рис. 2.16: Вопрос №16

Выполнено наполовину методом “научного тыка”(рис. 2.17).

Мы рассказали, что можно проверить код возврата внешней программы прямо в конструкции `if` при помощи `if "program options arguments"` (действия внутри `if` выполняются, если программа закончилась с кодом 0). Однако это не всегда правда! Если запуск внешней программы выводит что-то в `stdout`, то в проверку `if` поступит именно этот вывод, а не код возврата! Вы можете убедиться в этом, написав простой `bash`-скрипт с использованием, например, `if `pwd``.

Однако как быть, если хочется всё-таки запустить программу `program`, которая пишет что-то в `stdout` и потом выполнить какие-то действия если её код возврата равен 0? Выберите все верные утверждения или правильно работающие конструкции `if`.

Примечание: во всех вариантах ответов, где есть кавычка, используется именно косая кавычка (`'`), а не обычная (`"`) или двойная (`"`).

Выберите все подходящие ответы из списка

☒ Хорошие новости, верно!

Верно решили 21 426 учащихся
Из всех попыток 20% верных

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.

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

Следующий шаг

Решить снова

Ваши решения Вы получили: 1 балл из 1

Рис. 2.17: Вопрос №17

Важно обратить на область определения переменных в функциях(рис. 2.18).

3.4 Скрипты на `bash`: разное 9 из 10 шагов пройдено 9 из 14 баллов получено

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [functions1.sh](#), [functions2.sh](#).

Посмотрите на функцию из `bash`-скрипта:

```
counter () # takes one argument
{
    local let "c1+=1"
    let "c2+=${1}*2"
}
```

Впишите в форму ниже строку, которую выведет на экран команда `echo "counters are $c1 and $c2"` если она находится в скрипте после десяти вызовов функции `counter` с параметрами сначала 1, затем 2, затем 3 и т.д., последний вызов с параметром 10.

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.

Напишите текст

☒ Верно.

Верно решили 20 009 учащихся
Из всех попыток 28% верных

Рис. 2.18: Вопрос №18

Алгоритм дан, но я использовал вместо вычитания остаток от деления. Тоже работает (даже эффективнее)(рис. 2.19).

Напишите скрипт на `bash`, который будет искать наибольший общий делитель (НОД, greatest common divisor, GCD) двух чисел.

При запуске ваш скрипт не должен ничего писать на экран, а просто ждет ввода двух натуральных чисел через пробел (для этого можно использовать `read` и указать ему две переменные – см. пример в видеофрагменте). После ввода чисел скрипт считает их НОД и выводит на экран сообщение "GCD is <посчитанное значение>", например, для чисел 15 и 25 это будет "GCD is 5". После этого скрипт опять входит в режим ожидания двух натуральных чисел. Если в какой-то момент работы пользователь ввел вместо этого пустую строку, то нужно написать на экран `bye` и закончить свою работу.

Вычисление НОД не сложно реализовать с помощью [алгоритма Евклида](#). Вам нужно написать функцию `gcd`, которая принимает на вход два аргумента (назовем их `M` и `N`). Если аргументы равны, то мы нашли НОД – он равен `M` (или `N`), нужно вывести соответствующее сообщение на экран (см. выше). Иначе нужно сравнить аргументы между собой. Если `M` больше `N`, то запускаем ту же функцию `gcd`, но в качестве первого аргумента передаем `(M-N)`, а в качестве 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
18 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 отличие от `exit` эта команда завершит только функцию, а не выполнение всего скрипта целиком. Однако в данной задаче можно обойтись и без использования `return`!

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

Напишите программу. Тестируется через `stdin → stdout`

Так точно!

Верно решили 18 148 учащихся
 Из всех попыток 35% верных

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

Рис. 2.19: Вопрос №19

Написать калькулятор...(рис. 2.20).

Напишите программу. Тестируется через `stdin → stdout`

Всё правильно.

Верно решили 16 980 учащихся
 Из всех попыток 36% верных

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

```
1 #!/bin/bash
2 while [[ True ]]
3 do
4     read birinchi amal ikkinchi
5     if [[ $birinchi == "exit" ]]
6     then
7         echo "bye"
8         break
9     elif [[ "$birinchi" =~ ^[0-9]+$ && "$ikkinchi" =~ ^[0-9]+$ ]]
10    then
11        echo "error"
12        break
13    else
14        case $amal in
15            "+") let "result = birinchi + ikkinchi";;
16            "-") let "result = birinchi - ikkinchi";;
17            "/" ) let "result = birinchi / ikkinchi";;
18            "*" ) let "result = birinchi * ikkinchi";;
19            "%" ) let "result = birinchi % ikkinchi";;
20            "**") let "result = birinchi ** ikkinchi";;
21            *) echo "error" ; break ;;
22        esac
23        echo "$result"
24    fi
25 done
```

Следующий шаг

Решить снова

Рис. 2.20: Вопрос №20

Разница команд в учете и НЕ учете регистра при поиске(рис. 2.21).

Пусть в директории `/home/b1` лежат файлы `Star_Wars.avi`, `star_trek OST.mp3`, `STARS.txt`, `stardust.mpeg`, `Eddard_Stark_biography.txt`.

Отметьте все файлы, которые **найдет** команда `find /home/b1 -iname "star*"`, но **НЕ найдет** команда `find /home/b1 -name "star*"` ?

Выберите все подходящие ответы из списка

Верно решили 20 547 учащихся
Из всех попыток 36% верных

☒ Так точно!

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

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

👍 731 💬 232 Шаг 3 Следующий шаг >

Рис. 2.21: Вопрос №21

Проще воспользоваться методом исключения. Команды точно не работают одинаково и точно их результат не остается неизменным. Опции `path` и `name` используются не только для директорий и файлов соответственно(рис. 2.22).

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

☒ Так точно!

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](#).

- ☐ Опция `-path` используется только для поиска директорий, а `-name` только для поиска файлов
- ☒ В некоторых случаях `find c -name` найдет больше файлов, чем `find c` с таким же запросом, но с `-path`
- ☒ 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
- ☐ Опции `-path` и `-name` всегда работают одинаково

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

Рис. 2.22: Вопрос №22

Минимальная глубина 2, максимальная 3. Вывод напрашивается сам(рис. 2.23).

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

☒ Всё правильно.

Верно решили 20 711 учащихся
Из всех попыток 41% верных

- ☐ Все три файла
- ☐ Только file1
- ☒ Все кроме file3
- ☐ Только file2

Рис. 2.23: Вопрос №23

В данной ситуации опции не изменяют размер файла относительно друг друга(рис. 2.24).

3.5 Продвинутый поиск и редактирование 8 из 13 шагов пройдено 4 из 10 баллов получено

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.

Верно решили 20 237 учащихся
Из всех попыток 41% верных

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

Следующий шаг

Решить снова

Рис. 2.24: Вопрос №24

Проверил на практике. Результат удивил(рис. 2.25).

Что произойдет, если в команде `sed -n "/[a-z]*/p" text.txt` не указывать опцию `-n`?

Выберите один вариант из списка

☒ Верно. Так держать!

Верно решили 19 784 учащихся
Из всех попыток 39% верных

☐ Будут выведены все строки файла text.txt, в которых есть только большие буквы латинского алфавита
☐ На экран будет выведено всё содержимое файла text.txt
☐ Появится сообщение об ошибке
☒ Каждая строчка будет выведена два раза

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

731 232 Шаг 11 Следующий шаг >

30 Комментариев 3 Решения Самые популярные ▾

Рис. 2.25: Вопрос №25

Изучение документации снимает эту проблему(рис. 2.26).

3.6 Строим графики в `gnuplot` 3 из 10 шагов пройдено 1 из 7 баллов получен

Вы можете скачать и попробовать применить `gnuplot` к файлу, который мы показали в видеофрагменте: [authors.txt](#).

Какую опцию нужно указать при запуске `gnuplot`, чтобы при его закрытии не были автоматически закрыты и все нарисованные в нём графики?

Выберите один вариант из списка

☒ Абсолютно точно.

Верно решили 18 785 учащихся
Из всех попыток 51% верных

☐ Графики и так не закрываются автоматически при закрытии `gnuplot`!
☒ `-p, -persist`
☐ `-raise`
☐ `-s, -show-plots-after-exit`

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл из 1

Рис. 2.26: Вопрос №26

Крайне очевидно(рис. 2.27).

Предположим у вас есть файл `data.csv` с двумя столбцами по 10 чисел в каждом. В первой строке не записаны названия столбцов, т.е. ряды данных начинаются прямо с первой строки. Вы запускаете `gnuplot` и вводите в него две команды:

```
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

☒ Всё правильно.

Верно решили 17 975 учащихся
Из всех попыток 32% верных

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 – первое значение из первого столбца, нарисовано 9 точек (точка из первой строки пропущена)
- ☐ Название "data.csv" using 1:2", нарисовано 10 точек
- ☐ Название "no name", нарисовано 10 точек
- ☒ Название – первое значение из второго столбца, нарисовано 9 точек (точка из первой строки пропущена)
- ☐ Название – первое значение из второго столбца, нарисовано 10 точек

Рис. 2.27: Вопрос №27

Необходимо установить права доступа 764. Перечислены все варианты. Сложность вызвал факт, что нельзя просто указать права доступа используя символы `rwX`-(рис. 2.28).

Выберите все подходящие ответы из списка

☒ Отлично!

Верно решили 16 484 учащихся
Из всех попыток 21% верных

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 a+wx file.txt; chmod o-wx file.txt; chmod g-x file.txt`
- ☒ `chmod ug+w file.txt; chmod u+x file.txt`
- ☐ `chmod o-wx file.txt; chmod g-x file.txt; chmod a+wx file.txt`
- ☐ `chmod 467 file.txt`
- ☒ `chmod 764 file.txt`
- ☐ `chmod rwxrw-r-- file.txt`

Следующий шаг

Решить снова

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

Рис. 2.28: Вопрос №28

Внимательно смотрим на примечание и права доступа(рис. 2.29).

Дело в том, что если создать при помощи `sudo` файл с правами `rw-r--r--` в директории, которая принадлежит пользователю, то возникнет любопытная ситуация. С одной стороны пользователь может удалить этот файл (т.к. ему разрешено удалять **все** файлы внутри его директории) и может прочитать его содержимое (т.к. право `r` у файла установлено для всех), с другой стороны он не может этот файл редактировать (т.к. право `w` у файла есть только для **root**). При этом некоторые "умные" редакторы, например, **vim** позволяют даже редактировать этот файл, но делают они это своеобразно: через удаление оригинала и создание копии уже с нужными правами (удалять мы можем, а раз можем читать, то и копию создать не сложно). Итого получается, что несмотря на права `rw-r--r--`, пользователь может сделать с этим файлом почти всё что угодно!

В случае же, когда речь идет о директории созданной **root**, ситуация будет проще: пользователь сможет посмотреть её содержимое (у него есть право `r`), но удалять и создавать файлы в ней не сможет (права `w` у него нет).

Важно отметить, что директории в *Linux* это в каком-то смысле *файлы*. Содержимое такого "файла" – это записи о файлах и поддиректориях этой директории (грубо говоря их названия). Таким образом, право `r` у директории дает возможность просматривать "записи", т.е. просматривать её состав. Право `w` у директории дает возможность удалять/добавлять новые "записи", т.е. удалять/создавать файлы/поддиректории в ней.

На самом деле и это еще не всё. Существует так называемый **sticky bit** (атрибут файла или директории), выставление которого меняет описанное выше поведение. Файлы (или директории) с таким атрибутом сможет удалить только их владелец вне зависимости от прав, установленных у директории, в которой эти файлы (или директории) лежат!

Отдельное спасибо слушателю курса **Alexey Antipovsky** за помощь в оформлении **Примечания 2!**

Выберите все подходящие ответы из списка

☒ 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](#).

Верно решили **14 683** учащихся

Из всех попыток **15%** верных

☒ `sudo chown user:group dir`
☒ `sudo chmod o+w dir`
☐ `sudo chmod o+x dir`
☒ `sudo chmod a+w dir`
☐ `sudo chown :group dir`
☒ `sudo chown user dir`

Рис. 2.29: Вопрос №29

Сложностью является размер слов в байтах. Ведь в документации написано, что даётся размер строки(рис. 2.30).

Отметьте какие характеристики файла можно посчитать с использованием команды `wc`.

Выберите все подходящие ответы из списка

☒ 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](#).

Верно решили **17 158** учащихся

Из всех попыток **21%** верных

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

Следующий шаг

Решить снова

Ваши решения Вы получили: **1 балл** из 1

👍 1717

👏 107

Шаг 7

Следующий шаг >

Рис. 2.30: Вопрос №30

Посмотреть опции из видеоролика(рис. 2.31).

Впишите в форму ниже команду, которая выведет сколько места на диске занимает текущая директория (при этом **размер** нужно вывести в **удобном для чтения формате** (например, вместо `2848` байт надо вывести `2.8K`) и **больше** на экран выводить **ничего не** нужно). В команде указывайте **только необходимые** для выполнения задания **опции и аргументы**, лишних опций указывать не нужно!

Пример: если в текущей директории есть два файла по `800` Кбайт и две поддиректории в каждой из которой лежит по файлу в `400` Кбайт, то загаданная команда должна вывести на экран одно число: `2.4M` (также на экране может быть выведен еще и символ `"`, обозначающий, что это размер именно текущей директории).

Напишите текст

✓ Прекрасный ответ.

Верно решил **16 381** учащихся
Из всех попыток **53%** верных

`du -hs`

Следующий шаг

Решить снова

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

👍 1717 🗨 107 Шаг 8

Следующий шаг ➤

Рис. 2.31: Вопрос №31

Использовать знание команд на максимум(рис. 2.32).

Впишите в форму ниже максимально короткую команду (т.е. в которой минимально возможное число символов), которая позволит создать в текущей директории 3 поддиректории с именами `dir1`, `dir2`, `dir3`.

Если вы придумали команду, которая выполняет эту задачу, а система проверки сообщает вам "Incorrect"/"Неверно", то скорее всего вы придумали не самую короткую команду из возможных!

Напишите текст

✓ Правильно, молодец!

Верно решили **16 720** учащихся
Из всех попыток **40%** верных

`mkdir dir{1..3}`

Следующий шаг

Решить снова

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

👍 1717 🗨 107 Шаг 10

Следующий шаг ➤

Рис. 2.32: Вопрос №32

3 Вывод

В ходе выполнения лабораторной работы я рассмотрел продвинутые темы