

Внешний курс

Раздел 3

Полякова Ю.А.

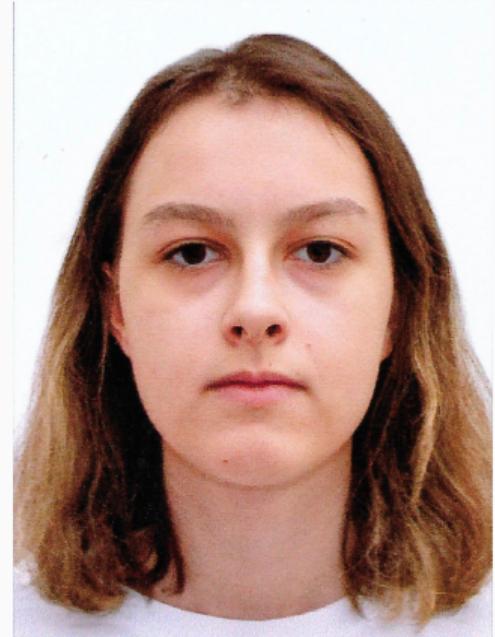
28 февраля 2007

Российский университет дружбы народов, Москва, Россия

Преподаватель Кулябов Д. С., д.ф.-м.н., профессор

Информация

- Полякова Юлия Александровна
- Студент
- Российский университет дружбы народов
- yulya.polyakova.07@mail.ru
- <https://github.com/JuliaMaffin123>



Вводная часть

- Прохождение этого внешнего курса даст полезные необходимые в дальнейшем базовые знания по работе с Linux

- Текстовый редактор vim
- Скрипты на bash
- Графики в gnuplot

- Цель: Получить базовые знания по работе с редактором, командными файлами и gnuplot
- Задачи:
 - Изучить теорию
 - Выполнить тестовые и интерактивные задания в курсе

- редактор gedit
- терминал
- stepik

Выполнение раздела

Часть “Текстовый редактор vim”

Шаг 5

: - переход в режим команд, q - команда выхода, Enter - отправить команду выхода.

The screenshot shows a user interface for a programming challenge. At the top, there is a navigation bar with icons for back, forward, search, and help, followed by a user icon and the number '0'. Below the bar, a progress message reads: '3.1 Текстовый редактор vim 12 из 12 шагов пройдено 7 из 7 баллов получено'. The main content area contains a question: '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.' Below the question, a list of options is presented:

- "Q"
- "Ctrl", затем "x"
- "q"
- ":"; затем "q", затем "Enter"
- ":"; затем "q"

Below the list are two buttons: 'Следующий шаг' (Next step) and 'Решить снова' (Solve again). A green box on the right indicates the result: 'Верно решили 32 523 учащихся' and 'Из всех попыток 69% верных'.

At the bottom, there is a footer with the text 'Ваши решения' and 'Вы получили: 1 балл'.

Рис. 1: 3.1 Шаг 5

Шаг 7

Логично, что W перемещается по большим словам, а w - с меньшей дистанцией, поэтому сразу логически понятны все верные ответы.

3.1 Текстовый редактор **txt**. 12 из 12 шагов проходено. 3 из 7 баллов получено.

При перимакции в **txt** по словам" есть небольшая разница в том, используемы маленькая (и, и, и) или большая (W, E, B) буквы. Первые перимакают нас по "словам" (word), а вторые по "большим словам" (WORD). Посмотрите справку по этим перимакциям и разберитесь, в чем заключается разница между word и WORD.

А для того, чтобы убедиться, что вы разобрались, отметьте **неко** все первые утверждения про следующую строку:

Strategic TEXT Is here... 2x YES!

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

Подсказка: чтобы вызвать **шиф-справку** по, например, перимакции И, нужно открыть меню и ввести команду :help И. Вы попадаете в то место справки, где описано это перимакение, а так как все перимакции скрыты подряд, то дальше по тексту ищете и вика можно прочитать и про И и про Б, и сама гланкон, про word и WORD. Кроме того, можно вызвать сразу справку по термину word при помощи :help word, чтобы закрыть справку, нужно ввести команду :q.

Выполните все подразумевающие ответы из списка

✓ Всё правильно.

Вы решите сложную задачу, задававшуюся! Вы можете поговорить с остальными учениками в [комментариях](#), ответя на их вопросы, или спросить свое решение с другими на [форуме решений](#).

✓ Чтобы вставить в конец строки, нужно свидетельствовать меньше кавычкой на W, чем на E

✓ В этой строке 9 'иско' (нет)

В этой строке 9 'Чтобы счесть слово' (WORD)

В этой строке 3 'иско' (нет)

Нужно не только на W, но и на E переместить курсор на "

Нужно не только на W, но и на E переместить курсор на "

Рис. 2: 3.1 Шаг 7

Шаг 9

Проанализировав каждую команду, можно понять, какие из них позволяют добиться этой строчки.

The screenshot shows a text editor window with the following content:

3.1 Текстовый редактор или 12 из 12 шагов пройдено 7 из 7 шагов получено

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.

Какой(ий) из предложенных ниже наборов клавиш выполнит 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 или, включен нормальный режим работы и курсор находится в самом начале строки.

Выберите все подходящие ответы из списка

Всё правильно.

Вы решите эти задачи, отправив нам Ваше решение в [помощь эксперту](#), чтобы
не як вороти, или привяжите Ваше решение к другому решению в [форуме решений](#).

Было решено 23 655 участниками
Из всех попыток 16% верных

✓ `1234567890`
✓ `d2d2eefour four <Esc>`
✓ `d3f3rre four four four f4ve<Esc>`
✓ `d3mmyd3r`
✓ `d3mmyFp`
✓ `d2d2ffour four <Esc>`

[Следующий шаг](#) [Решить снова](#)

Помощник эксперта Вы выполнили 1 из 10

Рис. 3: 3.1 Шаг 9

Шаг 11

Довольно простая команда, заменяющая слова.

The screenshot shows a challenge interface for a text editor. At the top, there's a navigation bar with icons for back, forward, search, and help, followed by a progress bar indicating "3.1 Текстовый редактор vim" has been completed. Below the progress bar, the text "12 из 12 шагов пройдено" and "7 из 7 баллов получено" are displayed. A green button labeled "Попытка" is visible on the right.

The main content area contains a text editor window with the following text:

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

Below the text editor, there are two buttons: "Следующий шаг" (Next Step) and "Решить снова" (Solve Again). At the bottom of the interface, a message states "Верно решил 24 631 учащийся" and "Из всех попыток 57% верных".

At the very bottom, a footer bar displays "Ваши решения" and "Вы получили: 2 балла".

Рис. 4: 3.1 Шаг 11

Шаг 12

После исследования этого режима в лабораторной работе основного курса сразу удалось определить ответы.

The screenshot shows a Moodle-based quiz interface. At the top, there is a navigation bar with icons for back, forward, search, and help, along with a user icon and a counter of 0 notifications. Below the navigation bar, a progress message reads: "3.1 Текстовый редактор vim 12 из 12 шагов пройдено 7 из 7 баллов получено". The main content area contains a question about the Visual mode in Vim. The question text is: "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!". Below the question, there is a note: "Чтобы убедиться, что вы разобрались с этим режимом работы, отметьте, пожалуйста, **все верные** утверждения из списка ниже." A tip below the note says: "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." A callout box highlights the instruction "Выберите все подходящие ответы из списка". The correct answer is marked with a green checkmark and labeled "Правильно". A note below the correct answer says: "Вы решили сложную задачу, поздравляем! Вы можете помочь остальным ученикам в [комментариях](#), ответив на их вопросы, или сравнить свой решение с другими на [форуме решений](#)". A green box at the top right indicates "Всего решено 21 497 учащихся" and "Из всех попыток 29% верных". The list of correct answers includes: "Режим выделения открывается при помощи команды visual", "Режим выделения открывается из нормального режима по нажатию "v", "В режиме выделения можно использовать команды перемещения (например, W, e, S, и др.)", "В режиме выделения можно использовать команды d (удалить) и u (копировать)", "Выйти из режима выделения можно, нажав клавишу Esc два раза", and "Когда вы находитесь в режиме выделения, внизу редактора горит надпись – VISUAL – (или – ВИЗУАЛЬНЫЙ РЕЖИМ –)". At the bottom, there are two buttons: "Следующий шаг" and "Решить снова". A note at the very bottom says: "Ваше решение Вы получили 2 балла".

Рис. 5: 3.1 Шаг 12

Часть “Скрипты на bash: основы”

Шаг 3

Самый логичный вариант.

The screenshot shows a user interface for a programming challenge. At the top, there's a navigation bar with icons for back, forward, search, and exit, along with a progress indicator showing "3.2 Скрипты на bash: основы 10 из 10 шагов пройдено 6 из 6 баллов получено". Below the navigation bar is a text block containing the following text:

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 попробуете при помощи стрелочек вверх/вниз перемещаться по истории набранных команд, то команды из какого набора(ов) будут появляться?

Below this text is a section titled "Выберите один вариант из списка" (Select one option from the list). It contains the following text: "Верно." (Correct) with a green checkmark icon. To the right of this text is a green box containing the following statistics: "Верно решили 30 266 учащихся" (Correctly solved by 30 266 students) and "Из всех попыток 65% верных" (Of all attempts, 65% were correct).

Below the "Correct" message is a list of five options, each preceded by a radio button:

- Из наборов А и С
- Только из набора В
- Только из набора А
- Из наборов В и С
- Только из набора С

At the bottom of the interface are two buttons: "Следующий шаг" (Next step) in green and "Решить снова" (Solve again) in white. At the very bottom, there's a status bar with the text "Ваши решения" (Your solutions) and "Вы получили: 1 балла" (You got: 1 point).

Рис. 6: 3.2 Шаг 3

Шаг 5

Мы переходим в `/home/bi/`, там создается файл. Значит путь: `/home/bi/имя файла`

The screenshot shows a user interface for an online programming exercise. At the top, there is a navigation bar with icons for back, forward, search, and help, followed by a progress indicator showing "3.2 Скрипты на bash: основы 10 из 10 шагов пройдено 6 из 6 баллов получено". Below the progress bar, a message says "Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [script1.sh](#), [script2.sh](#)". A note below it states: "Предположим, что вы находитесь в директории `/home/bi/Documents/` и запускаете в ней скрипт следующего содержания:" followed by the script code:

```
#!/bin/bash
cd /home/bi/
touch file1.txt
cd /home/bi/Desktop/
```

A question follows: "Как будет выглядеть **абсолютный путь** до созданного файла `file1.txt` по окончанию работы скрипта?" Below this, a section titled "Выберите один вариант из списка" contains a green checkmark icon and the text "Хорошая работа.". To the right, a statistics box shows "Верно решили 29 905 учащихся" and "Из всех попыток 76% верных". A list of four options is provided:

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

At the bottom, there are two buttons: "Следующий шаг" (Next step) and "Решить снова" (Solve again). A footer at the bottom of the page says "Ваши решения Вы получили: 1 балл".

Рис. 7: 3.2 Шаг 5

Шаг 7

В именах можно использовать буквы, нижние подчеркивания и цифры.

3.2 Скрипты на bash: основы 10 из 10 шагов пройдено 6 из 6 баллов получено

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 нажмите кнопку "Отправить"/"Submit".

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 свое решение с другими на [форуме решений](#).

variable_123
 __variable
 vari-able
 var@ible
 vari/able
 variab\$\$le
 variable123

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл

Рис. 8: 3.2 Шаг 7

Шаг 10

Здесь виден и ответ.

3.2 Скрипты на bash: основы 10 из 10 шагов пройдено 6 из 6 баллов получено

Вы можете скачать и изучить скрипт, который мы показали в видеофрагменте: [arguments.sh](#).

Напишите скрипт на bash, который принимает на вход два аргумента и выводит на экран строку следующего вида:

```
Arguments are: $1=первый_аргумент $2=второй_аргумент
```

Например, если ваш скрипт называется `./script.sh`, то при запуске его `./script.sh one two` на экране должно появиться:

```
Arguments are: $1=one $2=two
```

а при запуске `./script.sh three four` будет:

```
Arguments are: $1=three $2=four
```

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

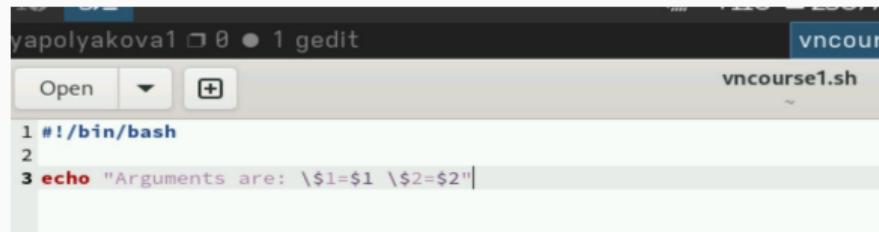
Хорошая работа.

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

```
1 echo "Arguments are: \$1=\$1 \$2=\$2"
2
3
4
```

Рис. 9: 3.2 Шаг 10, формулировка задания

Шаг 10, командный файл



The screenshot shows a terminal window titled 'yapolyakova1' with a sub-terminal window labeled 'gedit'. The file being edited is named 'vncourse1.sh'. The code in the editor is:

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

Рис. 10: 3.2 Шаг 10, листинг

Шаг 10

Это проверка работы файла. Делаем его исполняемым и запускаем.

```
yapolyakova1@yapolyakova1:~$ chmod +x vncourse1.sh
yapolyakova1@yapolyakova1:~$ ./vncourse1.sh one two
Arguments are: $1=one $2=two
yapolyakova1@yapolyakova1:~$ █
```

Рис. 11: 3.2 Шаг 10

Часть “Скрипты на bash: ветвления и циклы”

Шаг 3

Здесь подошли все варианты.

The screenshot shows a web-based programming environment for a task titled "3.3 Скрипты на bash: ветвления и циклы". The task has been completed with 10 out of 10 points. The code provided is:

```
if [ ... ]
then
    echo "True"
fi
```

A note below the code says: "Вы можете вписать вместо "...", скобки [[]], и не забудьте про проблемы после [[] и перед [][]] любое из перечисленных ниже условий. Однако мы просим вас выбрать только те из них, при которых echo напечатает на экран True, вне зависимости от того, с какими параметрами был запущен ваш скрипт и какие в нем есть переменные."

Another note states: "Например, условие \$ -eq 8 подходит, т.к. ноль всегда разен нулю вне зависимости от аргументов и переменных внутри скрипта и на экран будет напечатано True. В то же время условие \$var1 -eq \$ не подходит, так как в переменной var1, как может быть, записан ноль (тогда будет напечатано False), так его может и не быть (тогда ничего напечатано не будет)."

A note at the bottom left says: "Примечание: если вы планируете проверить 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."

The user has selected the correct answer: "Хорошая работа." A green box indicates "Верно решено 23 158 учеников" (23,158 students solved it) and "Из всех попыток 16% верны" (16% of attempts were correct).

A yellow box says: "Вы решали сложную задачу, поздравляем! Вы можете помочь остальным учащимся в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другим на [форуме решений](#),"

The list of correct answers includes:
✓ \$ == 0
✓ -z
✓ \$var1 == \$var2 || \$var1 != \$var2
✓ !(4 <= 3)
✓ 5 -gt 5
✓ -e \$0

Рис. 12: 3.3 Шаг 3

Шаг 5

3 и 5 не проходят ни по одному из условий и идет в else

3.3 Скрипты на bash: ветвления и циклы 9 из 9 шагов пройдено 10 из 10 баллов получено

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

Абсолютно точно.

Верно решили 25 138 учащихся
Из всех попыток 64% верных

- Сначала two, потом four
- Сначала four, потом four
- Сначала four, потом one
- Сначала two, потом one

Рис. 13: 3.3 Шаг 5

Шаг 6

Здесь формулировка задания

3.3 Скрипты на bash: ветвления и циклы 9 из 9 шагов пройдено 10 из 10 баллов получено

Напишите скрипт на bash, который принимает на вход один аргумент (целое число от 0 до бесконечности), который будет обозначать число студентов в аудитории. В зависимости от значения числа нужно вывести разные сообщения.

Соответствие входа и выхода должно быть таким:

```
B --> No students  
1 --> 1 student  
2 --> 2 students  
3 --> 3 students  
4 --> 4 students  
5 и больше --> A lot of students
```

Примечание а): выводить нужно только строку справа, т.е. "->" выводить не нужно.
Примечание б): 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`

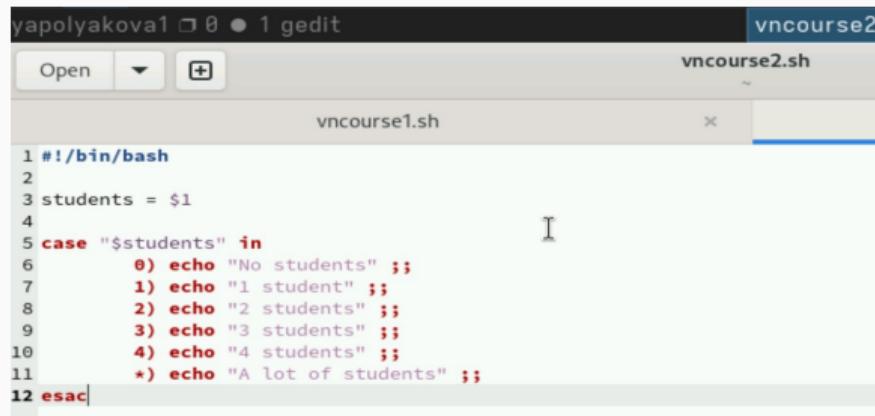
Верно решили 23 310 учащихся
Из всех попыток 38% верных

Верно.

Рис. 14: 3.3 Шаг 6, формулировка задания

Шаг 6

Командный файл, который решает задачу.



The screenshot shows a terminal window titled "yapolyakova1" with a status bar indicating 0 open files and 1 gedit process. The window contains two tabs: "vncourse2.sh" (active) and "vncourse1.sh". The code in "vncourse2.sh" is as follows:

```
1 #!/bin/bash
2
3 students = $1
4
5 case "$students" in
6     0) echo "No students" ;;
7     1) echo "1 student" ;;
8     2) echo "2 students" ;;
9     3) echo "3 students" ;;
10    4) echo "4 students" ;;
11    *) echo "A lot of students" ;;
12 esac
```

Рис. 15: 3.3 Шаг 6, командный файл

Шаг 6

Делаем файл исполняемым, проверяем с разными данными

```
yapolyakova1@yapolyakova1:~$ chmod +x vncourse2.sh
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 0
No students
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 1
1 student
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 2
2 students
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 3
3 students
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 4
4 students
yapolyakova1@yapolyakova1:~$ ./vncourse2.sh 5
A lot of students
yapolyakova1@yapolyakova1:~$
```

Рис. 16: 3.3 Шаг 6, проверка работы

Шаг 8

Ответ 5 раз “старт” и 4 раза “финиш”. Я проверила это создав и запустив командный файл.

3.3 Скрипты на bash: ветвления и циклы 9 из 9 шагов пройдено 10 из 10 баллов получено

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

Верно решили 24 582 учащихся
Из всех попыток 45% верных

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

Отправить

Ваше решение Вы получили: 1 балл

Рис. 17: 3.3 Шаг 8

Шаг 9

Здесь задание

3.3 Скрипты на bash: ветвления и циклы 9 из 9 шагов пройдено 10 из 10 баллов получено

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 (справа 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:

Nº1

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

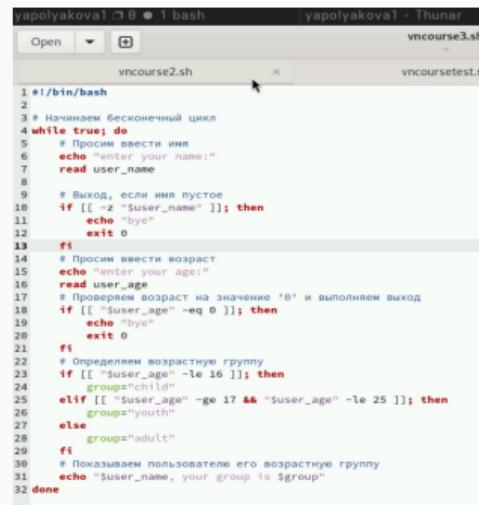
Nº2:

```
./script.sh
enter your name:
```

Рис. 18: 3.3 Шаг 9, формулировка задания

Шаг 9

Командный файл-ответ



The screenshot shows a terminal window titled "yapolyakova1 - Thunar". The window contains two tabs: "vncourse2.sh" and "vncourse3.sh". The "vncourse2.sh" tab is active and displays the following Bash script:

```
1#!/bin/bash
2
3# Начинаем бесконечный цикл
4while true; do
5    # Просим ввести имя
6    echo "enter your name:"
7    read user_name
8
9    # Выход, если имя пустое
10   if [[ -z "$user_name" ]]; then
11       echo "bye"
12       exit 0
13   fi
14   # Просим ввести возраст
15   echo "enter your age:"
16   read user_age
17   # Проверяем возраст на значение "0" и выполняем выход
18   if [[ "$user_age" -eq 0 ]]; then
19       echo "bye"
20       exit 0
21   fi
22   # Определяем возрастную группу
23   if [[ "$user_age" -le 16 ]]; then
24       group="child"
25   elif [[ "$user_age" -ge 17 && "$user_age" -le 25 ]]; then
26       group="youth"
27   else
28       group="adult"
29   fi
30   # Показываем пользователю его возрастную группу
31   echo "$user_name, your group is $group"
32 done
```

Рис. 19: 3.3 Шаг 9, командный файл

Шаг 9

Открываем доступ на исполнение, запускаем с разными данными.

```
yapolyakova1@yapolyakova:~$ chmod +x vncourse3.sh
yapolyakova1@yapolyakova:~$ ./vncourse3.sh
enter your name:
Julia
enter your age:
18
Julia, your group is youth
enter your name:
masha
enter your age:
3
masha, your group is child
enter your name:
misha
enter your age:
50
misha, your group is adult
enter your name:
0
enter your age:
0
bye
yapolyakova1@yapolyakova:~$
```

Рис. 20: 3.3 Шаг 9, проверка работы

Часть “Скрипты на bash: разное”

Шаг 3

В первом неподходящем варианте + и = не в том порядке, во втором - нет let, остальные подходят.

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

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

Какие(ая) из предложенных ниже инструкций увеличивает 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 содержание символов `$` тексты могут изменяться при копировании – не забудьте отредактировать их в соответствии с изображением на экране. Это связано с особенностями написания `$` 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

Хорошая работа.

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

let a=a+b
 let "a+=b"
 a+=\\$b
 let 'a+=b'
 let a=\\$a+\\$b

Следующий шаг **Решить снова**

Берно решили 22 116 учащиков
Из всех попыток 20% верных

Рис. 21: 3.4 Шаг 3

Шаг 5

Мы перешли командой cd в директорию, значит pwd выведет именно ее путь.

The screenshot shows a programming challenge interface. At the top, there are navigation icons (back, forward, search, etc.) and a progress bar indicating 10 из 10 шагов пройдено / 14 из 14 баллов получено. The main text asks: "Вы можете скачать и изучить скрипт, который мы показали в видеофрагменте: [programs.sh](#). Пусть вы находитесь в директории /home/b1/Documents/ и запускаете в ней скрипт следующего содержания:". Below this is a code snippet:

```
#!/bin/bash
cd /home/b1/
echo "$pwd"
```

The question below the code asks: "Что в этом случае выведет команда echo на экран?". A green box at the bottom right indicates: "Верно решили 23 677 учащихся Из всех попыток 51% верных".

Выберите один вариант из списка

Отличное решение!

/home/b1/Documents
 /home/b1
 pwd
 `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

Рис. 22: 3.4 Шаг 5

Шаг 6

Программа сначала должна вывести результат на экран или в файл, а потом мы сможем работать с кодом возврата.

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

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` поступят именно этот вывод, а не код возврата! Вы можете убедиться в этом, написав простой bash-скрипт с использованием, например, `if `ps``.

Однако как быть, если хочется всё-таки запустить программу `program`, которая пишет что-то в `stdout` и потом выполнить какие-то действия если ее 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

Всё правильно.

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

`if [['program' -eq 0]]`

`if 'program > some_file.txt'`

Ничего сделать нельзя

Сначала запустить `program`, затем `if [[$? -eq 0]]`

Сначала `var= program`, затем `if [[$var -eq 0]]`

[Следующий шаг](#) [Решить снова](#)

Вашим решением Вы получили: 1 балл

Рис. 23: 3.4 Шаг 6

Шаг 8

Эту задачу я проверила у себя, затем скопировала строку, которая выводится.

3.4 Скрипты на bash: разное 10 из 10 шагов пройдено 14 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

Хорошие новости, верно!

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

counters are and 110

Верно решили 20 009 учащихся
Из всех попыток 28% верных

Рис. 24: 3.4 Шаг 8

Шаг 9

Формулировка задания.

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

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 пользователя ввел вместо этого пустую строку, то нужно написать на экран «bye» и закончить свою работу.

Вычисление НОД несложно 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 выводить соответствующее сообщение на экран (см. выше). Иначе нужно сравнивать аргументы между собой. Если M большие N , то запускаем функцию `gcd`, но в качестве первого аргумента передаем MN , 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  
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`!

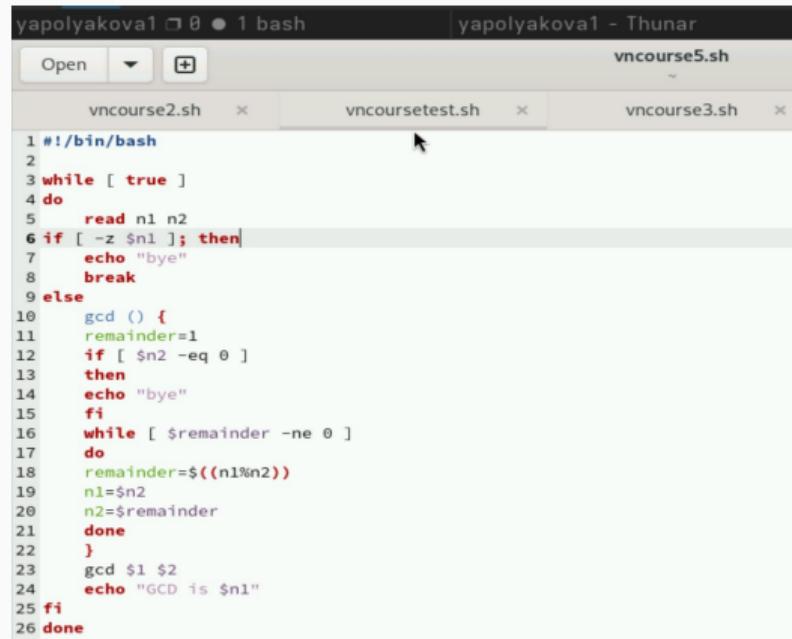
Примечание 2: для завершения работы функции в произвольном месте, можно использовать инструкцию `return` (все инструкции функции после `return` выполняться не будут). В отличии от `exit` эта команда завершит только функцию, а не выполнение всего скрипта целиком. Однако в данной задаче можно обойтись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

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

Рис. 25: 3.4 Шаг 9, формулировка

Шаг 9

Листинг командного файла.



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

Рис. 26: 3.4 Шаг 9, командный файл

Шаг 9

Как обычно, открываем доступ на исполнение, запускаем, тестируем.

```
yapolyakova1@yapolyakova1:~$ chmod +x vncourse5.sh
yapolyakova1@yapolyakova1:~$ ./vncourse5.sh
10 15
GCD is 5
7 3
GCD is 1

bye
yapolyakova1@yapolyakova1:~$ █
```

Рис. 27: 3.4 Шаг 9, проверка работы

Шаг 10

Формулировка задания.

Напишите калькулятор на bash. При запуске ваш скрипт должен ожидать ввода пользователем команды (при этом на экран выводить ничего не нужно). Команды могут быть трех типов:

- Слово "exit". В этом случае скрипт должен вывести на экран слово "bye" и завершить работу.
- Три аргумента через пробел – первый операнд (целое число), операция (один из "+", "-", "*", "/", "%", "**") и второй операнд (целое число). В этом случае нужно произвести указанную операцию над заданными числами и вывести результат на экран. После этого переходи в режим ожидания новой команды.
- Любая другая команда из одного аргумента или из трех аргументов, но с операцией не из списка. В этом случае нужно вывести на экран слово "етого" и завершить работу.

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:

```
18 + 1
2 ** 10
exit
```

то на экране будет:

```
11
1024
bye
```

Если же на вход поступит следующий файл:

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

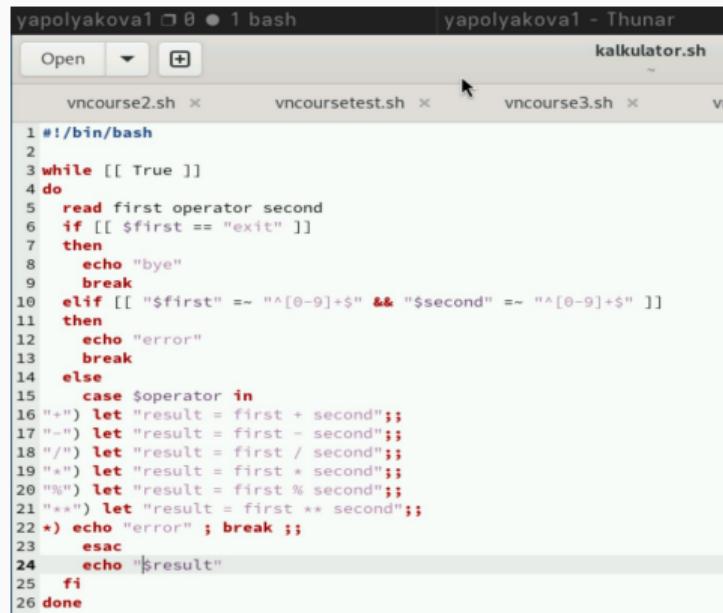
то на экране будет:

```
-2
етого
```

Рис. 28: 3.4 Шаг 10, формулировка

Шаг 10

Листинг командного файла калькулятора.



The screenshot shows a terminal window titled 'yapolyakova1' with a process ID of 0. The window contains the script 'kalkulator.sh'. The script is a Bash script that reads two numbers and an operator from the user, performs the calculation, and prints the result. It includes error handling for invalid input and operators.

```
#!/bin/bash
while [[ True ]]
do
    read first operator second
    if [[ $first == "exit" ]]
    then
        echo "bye"
        break
    elif [[ "$first" =~ ^[0-9]+\$" && "$second" =~ ^[0-9]+\$" ]]
    then
        echo "error"
        break
    else
        case $operator in
            "+") let "result = first + second";;
            "-") let "result = first - second";;
            "/") let "result = first / second";;
            "*") let "result = first * second";;
            "%") let "result = first % second";;
            "**") let "result = first ** second";;
            *) echo "error" ; break ;;
        esac
        echo "$result"
    fi
done
```

Рис. 29: 3.4 Шаг 10, командный файл

Шаг 10

Как обычно, открываем доступ на исполнение, запускаем, тестируем.

```
yapolyakova1@yapolyakova1:~$ chmod +x kalkulator.sh
yapolyakova1@yapolyakova1:~$ ./kalkulator.sh
10 + 1
11
2 ** 10
1024
exit
bye
yapolyakova1@yapolyakova1:~$ ./kalkulator.sh
3 - 5
-2
2/10
error
yapolyakova1@yapolyakova1:~$
```

Рис. 30: 3.4 Шаг 10, проверка работы

Часть “Продвинутый поиск и редактирование”

Шаг 3

Первой команде не важен регистр, а второй важен (из-за опций `iname` и `name`)

The screenshot shows a programming challenge interface. At the top, there are navigation icons (back, forward, search, etc.) and a user icon. Below that, the title "3.5 Продвинутый поиск и редактирование" is displayed, along with progress information: "13 из 13 шагов пройдено" and "10 из 10 баллов получено".

The main text of the question reads:
Пусть в директории /home/bi 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/bi -iname "star*`, но **НЕ найдет** команда `find /home/bi -name "star*`?

Below the question, there is a list of files:
Eddard_Stark.biography.txt
star_trek_OST.mp3
stardust.mpeg
✓ STARS.txt
✓ Star_Wars.avi

A green checkmark next to "Всё получилось!" indicates the answer is correct. A green box at the bottom right shows statistics: "Верно решили 20 547 учащихся" and "Из всех попыток 36% верных".

At the bottom, there are two buttons: "Следующий шаг" and "Решить снова".

At the very bottom, it says "Ваши решения Вы получили: 1 балл".

Рис. 31: 3.5 Шаг 3

Шаг 4

Здесь можно понять, рассуждая логически, основываясь на материалах теории из видео.

The screenshot shows a user interface for a programming exercise. At the top, there is a navigation bar with icons for help, search, and other functions. Below it, a progress bar indicates '3.5 Продвинутый поиск и редактирование' is being worked on, with '13 из 13 шагов пройдено' and '10 из 10 баллов получено'. The main content area contains the following text:

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

Выберите все подходящие ответы из списка

Верно.

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

Верно решили 18 450 учащихся
Из всех попыток 22% верных

List of options (checkboxes):

- В некоторых случаях `find c -path` найдет меньше файлов, чем `find c` с таким же запросом, но с `-path`
- 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 `-path`, на `-path`, то результат поиска иногда может остаться таким же
- Опции `-path` и `-name` всегда работают одинаково

Buttons at the bottom:

Следующий шаг Решить снова

Ваши решения Вы получили: 1 балл

Рис. 32: 3.5 Шаг 4

Шаг 5

Глубина $/home/bi = 1$, $dir1 = 2$, $dir2 = 3$. Поэтому будет только первый и второй файлы.

The screenshot shows a user interface for a programming exercise. At the top, there is a navigation bar with icons for back, forward, search, and help, followed by a progress indicator: "3.5 Продвинутый поиск и редактирование 13 из 13 шагов пройдено 10 из 10 баллов получено". Below this is a message: "Предположим, что в директории /home/bi/ есть следующая структура файлов и поддиректорий:" followed by a file tree diagram:

```
graph TD; Root[/home/bi/] --> dir1[dir1]; dir1 --> file1[file1]; dir1 --> dir2[dir2]; dir2 --> file2[file2]; dir2 --> dir3[dir3]; dir3 --> file3[file3]
```

Below the tree, a question is displayed: "Какие(о) из трех файлов (file1, file2, file3) будут найдены по команде `find /home/bi -mindepth 2 -maxdepth 3 -name "file*"`?"

Below the question, a message says "Выберите один вариант из списка" and "Отлично!" with a green checkmark icon. To the right, a statistics box shows "Верно решили 20 711 учащихся" and "Из всех попыток 41% верных".

The list of options for the question is as follows:

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

At the bottom, there are two buttons: "Следующий шаг" (Next step) and "Решить снова" (Solve again). A footer at the very bottom says "Ваши решения Вы получили: 1 балл".

Рис. 33: 3.5 Шаг 5

Шаг 7

А - num строк конечного контекста после сопоставления, В - num строк начального контекста перед соотв. строками, С - num строк начального и конечного выходного контекста.

3.5 Продвинутый поиск и редактирование 13 из 13 шагов пройдено 10 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

Так точно!

Верно решили 20 237 учащихся
Из всех попыток 41% верных

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

Следующий шаг Решить снова

Ваше решение Вы получили: 1 балл

Рис. 34: 3.5 Шаг 7

Шаг 9

Первая буква в Ubuntu может быть большой или маленькой, перед словом должно быть любое сочетание по порядку из скобок левее. В неправильном варианте нет и

The screenshot shows a challenge interface from a platform like Codecademy or LeetCode. At the top, there's a navigation bar with icons for help, back, forward, and search, followed by a user icon and a '0' badge. Below the bar, the title '3.5 Продвинутый поиск и редактирование' is displayed, along with progress information: '13 из 13 шагов пройдено' and '10 из 10 баллов получено'. The main text of the challenge reads: 'Предположим, что в файле text.txt записаны строки, показанные среди вариантов ответа. Отметьте только те из них, которые выведет на экран команда grep -E "[xkLXKL]?[uU]buntu\$" text.txt.' Below this, a question asks to select all correct answers from a list. A green checkmark next to the first option indicates it is correct. A yellow box contains a congratulatory message: 'Вы решили сложную задачу, поздравляем! Вы можете помочь остальным ученикам в [комментариях](#), отвечая на их вопросы, или сравнить своё решение с другими на [форуме решений](#)'. The correct answers listed are: 'The best OS is Xubuntu', 'Linux is not always Ubuntu', 'Lubuntu is better than Ubuntu', 'Kubuntu', 'Hmm, XKLUbuntu', and 'I prefer Kubuntu'. At the bottom, there are two buttons: 'Следующий шаг' (Next step) and 'Решить снова' (Solve again). A note at the very bottom says 'Ваше решение: Вы получили: 2 балла'.

Рис. 35: 3.5 Шаг 9

Шаг 11

-п ключ подавляет автоматический вывод шаблона

The screenshot shows a user interface for a programming challenge. At the top, there is a navigation bar with icons for help, search, and user profile. Below it, a progress bar indicates "3.5 Продвинутый поиск и редактирование" has been completed, with "13 из 13 шагов пройдено" and "10 из 10 баллов получено". The main content area contains a question: "Что произойдет, если в команде `sed -n /[a-z]*p* text.txt` не указывать опцию `-n`?" Below the question, a message says "Выберите один вариант из списка" and "Всё правильно." A green checkmark icon is present. To the right, a box displays statistics: "Верно решили 19 784 учащихся" and "Из всех попыток 39% верных". The list of options includes:

- На экран ничего не напечатается
- Каждая строчка будет выведена два раза
- На экран будет выведено всё содержимое файла text.txt
- Будут выведены все строки файла text.txt, в которых есть только большие буквы латинского алфавита

At the bottom, there are two buttons: "Следующий шаг" (in a green box) and "Решить снова". A status bar at the bottom left says "Ваши решения" and "Вы получили: 1 балл".

Рис. 36: 3.5 Шаг 11

Шаг 12

В sed передается регулярное выражение замены s.

The screenshot shows the 'Search & Replace' dialog in Notepad++. The 'Search' tab is selected. In the 'Search mode' dropdown, 'Regular expression' is chosen. The 'Search for' field contains the regular expression `sed 's/[A-Z][A-Z]/abbreviation/g' input.txt modified.txt`. The 'Replace with' field is empty. Below the search fields, there is explanatory text about abbreviations and their handling by sed. At the bottom, there are buttons for 'Find Next' (F3), 'Replace' (R), 'Replace All' (Ctrl+R), and 'Close' (X). A status bar at the bottom indicates 'Выполнено 16 422 операции' (16,422 operations completed).

Рис. 37: 3.5 Шаг 12

Часть “Строим графики в gnuplot”

Шаг 3

Эта опция была показана в видео.

The screenshot shows a user interface for an educational platform. At the top, there is a navigation bar with icons for back, forward, search, and help, along with a user icon and a '0' badge. Below the navigation bar, the title of the task is displayed: '3.6 Строим графики в gnuplot 10 из 10 шагов пройдено 7 из 7 баллов получено'. The main content area contains a question: '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?'. Below the question, there is a section titled 'Выберите один вариант из списка' with a green checkmark next to the correct answer: '-p, --persist'. There are also three other options listed without checkmarks: '-s, --show-plots-after-exit', '-raise', and 'Графики и так не закрываются автоматически при закрытии gnuplot!'. At the bottom of this section, there are two buttons: 'Следующий шаг' (Next step) and 'Решить снова' (Solve again). To the right of the question, there is a green box containing the statistics: 'Верно решили 18 785 учащихся' and 'Из всех попыток 51% верных'. At the very bottom of the page, there is a footer with the text 'Ваши решения Вы получили 1 балл'.

Рис. 38: 3.6 Шаг 3

Шаг 5

Ответ основан на материалах из видео теории.

3.6 Строим графики в gnuplot 10 из 10 шагов пройдено 7 из 7 баллов получено

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 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](#).

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

[Следующий шаг](#) [Решить снова](#)

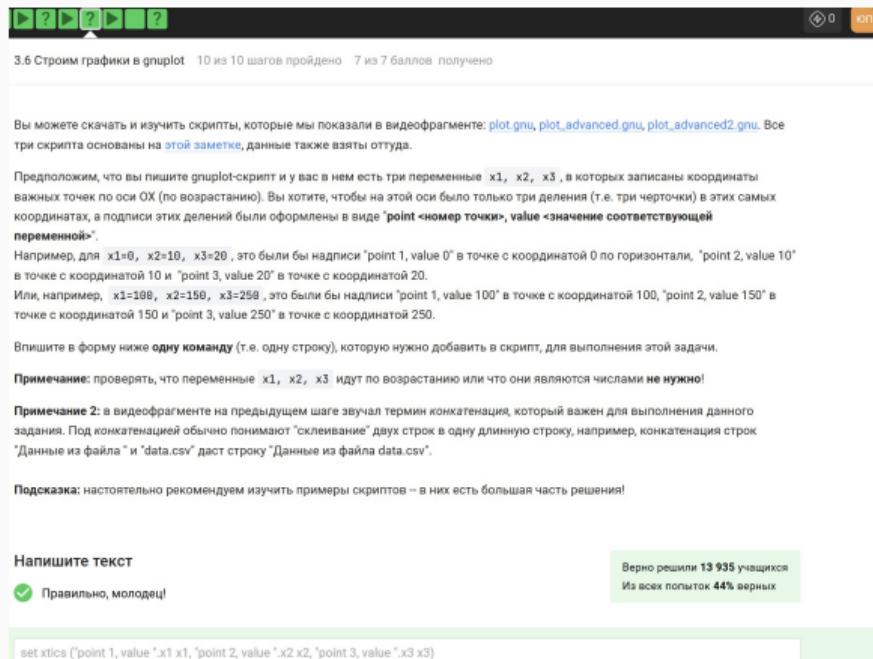
Ваше решение **Вы получили: 1 балл**

Верно решили 17 975 учащихся
Из всех попыток 32% верных

Рис. 39: 3.6 Шаг 5

Шаг 7

Сначала идет команда установки подписей, а потом в скобках подпись.



3.6 Строим графики в gnuplot 10 из 10 шагов пройдено 7 из 7 баллов получено

Вы можете скачать и изучить скрипты, которые мы показали в видеофрагменте: [plot.gnu](#), [plot_advanced.gnu](#), [plot_advanced2.gnu](#). Все три скрипта основаны на [этой заметке](#), данные также взяты оттуда.

Предположим, что вы пишите gnuplot-скрипт и у вас в нем есть три переменные `x1`, `x2`, `x3`, в которых записаны координаты важных точек по оси ОХ (по возрастанию). Вы хотите, чтобы на этой оси было только три деления (т.е. три черточки) в этих самых координатах, а подписи этих делений были оформлены в виде "point «номер точки», value «значение соответствующей переменной»".

Например, для `x1=0`, `x2=10`, `x3=20`, это были бы надписи "point 1, value 0" в точке с координатой 0 по горизонтали, "point 2, value 10" в точке с координатой 10 и "point 3, value 20" в точке с координатой 20.

Или, например, `x1=100`, `x2=150`, `x3=250`, это были бы надписи "point 1, value 100" в точке с координатой 100, "point 2, value 150" в точке с координатой 150 и "point 3, value 250" в точке с координатой 250.

Впишите в форму ниже **одну команду** (т.е. одну строку), которую нужно добавить в скрипт, для выполнения этой задачи.

Примечание: проверять, что переменные `x1`, `x2`, `x3` идут по возрастанию или что они являются числами **не нужно**!

Примечание 2: в видеофрагменте на предыдущем шаге звучал термин конкатенация, который важен для выполнения данного задания. Под конкатенацией обычно понимают "склеивание" двух строк в одну длинную строку, например, конкатенация строк "Данные из файла" и "data.csv" даст строку "Данные из файла data.csv".

Подсказка: настоятельно рекомендуем изучить примеры скриптов -- в них есть большая часть решения!

Напишите текст

Правильно, молодец!

Верно решили 13 935 учащихся
Из всех попыток 44% верных

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

Рис. 40: 3.6 Шаг 7

Шаг 10

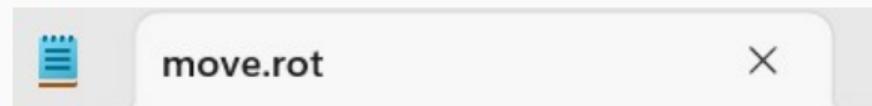
Формулировка задания.

The screenshot shows a user interface for an online assignment. At the top, there is a navigation bar with icons for back, forward, search, and help, along with a counter '0' and a button labeled 'ЮП' (YUP). Below the navigation bar, the title of the task is displayed: '3.6 Строим графики в gnuplot' followed by '10 из 10 шагов пройдено' and '7 из 7 баллов получено'. A message in the center instructs the user to download files if they were not downloaded previously, mentioning 'animated.gnu' and 'move.rot'. It also states that these files were used in a previous video fragment for creating a rotating plot. The user is asked to change the instructions in 'move.got' to achieve specific results: the graph should reflect relative to a horizontal plane, the image should rotate in the opposite direction, and the rotation speed should double. Below this, a note says to upload the modified 'move.got' file. A green checkmark icon indicates 'Хорошая работа.' (Good work). To the right, a box shows statistics: 'Верно решили 12 854 учащихся' (Correctly solved by 12,854 students) and 'Из всех попыток 47% верных' (Of all attempts, 47% are correct). At the bottom, there are buttons for 'Следующий шаг' (Next step) and 'Решить снова' (Solve again), along with a status bar showing 'Ваши решения' and 'Вы получили: 3 балла'.

Рис. 41: 3.6 Шаг 10, формулировка

Шаг 10

Меняем 10 на 350, ставим минусы у координат, уменьшаем паузу в 2 раза.



The screenshot shows a file editor window with a title bar containing a document icon, the text "move.rot", and a close button (X). Below the title bar is a menu bar with three items: "Файл" (File), "Изменить" (Edit), and "Просмотр" (View). The main area of the editor contains the following text:

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

Рис. 42: 3.6 Шаг 10, измененный файл

Вывод

Был полностью завершен третий раздел внешнего курса.