

Отчет 1,2 и 3

Linux

Силвен Макс Грегор Филс , НКАбд-03-22

Содержание

1 Цель работы	7
2 Теоретическое введение	8
3 Прохождение курса	10
3.1 Введение в Linux	10
3.2 Работа на сервере	25
3.3 Продвинутые темы	40
3.4 Сертификат	63
4 Выводы	65
Список литературы	66

Список иллюстраций

3.1	Профиль в Stepik	10
3.2	Ответ на вопрос	11
3.3	Ответ на вопрос	11
3.4	Ответ на вопрос	12
3.5	Ответ на вопрос	12
3.6	Ответ на вопрос	13
3.7	Ответ на вопрос	13
3.8	Ответ на вопрос	14
3.9	Ответ на вопрос	14
3.10	Ответ на вопрос	15
3.11	Ответ на вопрос	15
3.12	Ответ на вопрос	16
3.13	Ответ на вопрос	16
3.14	Ответ на вопрос	17
3.15	Ответ на вопрос	17
3.16	Ответ на вопрос	18
3.17	Ответ на вопрос	18
3.18	Ответ на вопрос	19
3.19	Ответ на вопрос	19
3.20	Ответ на вопрос	20
3.21	Ответ на вопрос	20
3.22	Ответ на вопрос	21
3.23	Ответ на вопрос	21
3.24	Ответ на вопрос	22
3.25	Ответ на вопрос	22
3.26	Ответ на вопрос	23
3.27	Ответ на вопрос	23
3.28	Ответ на вопрос	24
3.29	Ответ на вопрос	24
3.30	Ответ на вопрос	25
3.31	Ответ на вопрос	25
3.32	Ответ на вопрос	26
3.33	Ответ на вопрос	26
3.34	Ответ на вопрос	27
3.35	Ответ на вопрос	27
3.36	Ответ на вопрос	28
3.37	Ответ на вопрос	28

3.38 Ответ на вопрос	29
3.39 Ответ на вопрос	30
3.40 Ответ на вопрос	31
3.41 Ответ на вопрос	31
3.42 Ответ на вопрос	32
3.43 Ответ на вопрос	33
3.44 Ответ на вопрос	33
3.45 Ответ на вопрос	34
3.46 Ответ на вопрос	35
3.47 Ответ на вопрос	35
3.48 Ответ на вопрос	36
3.49 Ответ на вопрос	37
3.50 Ответ на вопрос	37
3.51 Ответ на вопрос	38
3.52 Ответ на вопрос	38
3.53 Ответ на вопрос	39
3.54 Ответ на вопрос	39
3.55 Ответ на вопрос	40
3.56 Ответ на вопрос	41
3.57 Ответ на вопрос	41
3.58 Ответ на вопрос	42
3.59 Ответ на вопрос	43
3.60 Ответ на вопрос	43
3.61 Ответ на вопрос	44
3.62 Ответ на вопрос	44
3.63 Ответ на вопрос	45
3.64 Ответ на вопрос	46
3.65 Ответ на вопрос	46
3.66 Ответ на вопрос	47
3.67 Ответ на вопрос	48
3.68 Ответ на вопрос	48
3.69 Ответ на вопрос	49
3.70 Ответ на вопрос	49
3.71 Ответ на вопрос	50
3.72 Ответ на вопрос	50
3.73 Ответ на вопрос	51
3.74 Ответ на вопрос	52
3.75 Ответ на вопрос	53
3.76 Ответ на вопрос	53
3.77 Ответ на вопрос	54
3.78 Ответ на вопрос	55
3.79 Ответ на вопрос	55
3.80 Ответ на вопрос	56
3.81 Ответ на вопрос	56

3.82 Ответ на вопрос	57
3.83 Ответ на вопрос	57
3.84 Ответ на вопрос	58
3.85 Ответ на вопрос	58
3.86 Ответ на вопрос	59
3.87 Ответ на вопрос	60
3.88 Ответ на вопрос	60
3.89 Ответ на вопрос	61
3.90 Ответ на вопрос	61
3.91 Ответ на вопрос	62
3.92 Ответ на вопрос	63
3.93 Итоги курса	63
3.94 Сертификат	64

Список таблиц

1 Цель работы

- Уметь пользоваться системой Linux

2 Теоретическое введение

- Linux – это семейство операционных систем (ОС), работающих на основе одноименного ядра. Нет одной операционной системы Linux, как, например, Windows или MacOS. Есть множество дистрибутивов (набор файлов, необходимых для установки ПО), выполняющих конкретные задачи.
- Где нужен Linux? - Веб-серверы Дистрибутивы Linux практически полностью захватили рынок веб-серверов. Согласно рейтингу аналитического агентства W3Techs, на Linux-серверах развернуты 75,1% сайтов. - Мобильные устройства ОС
- Android работает на ядре Linux, поэтому она используется и в мобильных устройствах. - Суперкомпьютеры Это специализированные вычислительные машины, превосходящие по своим техническим параметрам и скорости вычислений многие обычные компьютеры. Они уникальны, для многих из них требуется особая ОС, способная решать конкретные задачи. Благодаря открытому исходному коду разработчики могут видоизменять ОС так, чтобы она работала и на таких машинах.
- Игровые консоли Linux занял свою нишу и в игровых консолях, но ориентированных на эту ОС игр пока не так много. Компания Steam работает над исправлением ситуации – разрабатывает операционную систему SteamOS. Она будет поставляться вместе с игровой консолью Steam Machine.
- Устройства IoT и умная техника Многие из них созданы на основе Linux. Так, компания Samsung разработала операционную систему Tizen, LG – WebOS, а Panasonic и Philips используют FirefoxOS. - Авиация и транспорт Во встро-

енных компьютерах Tesla и машинах с автопилотами Google используется операционная система Linux. ПО для отслеживания трафика в США аналогично разработано на этой ОС, а администрация авиации Америки перешла на нее еще в 2006 году.

3 Прохождение курса

3.1 Введение в Linux

(рис. 3.1).

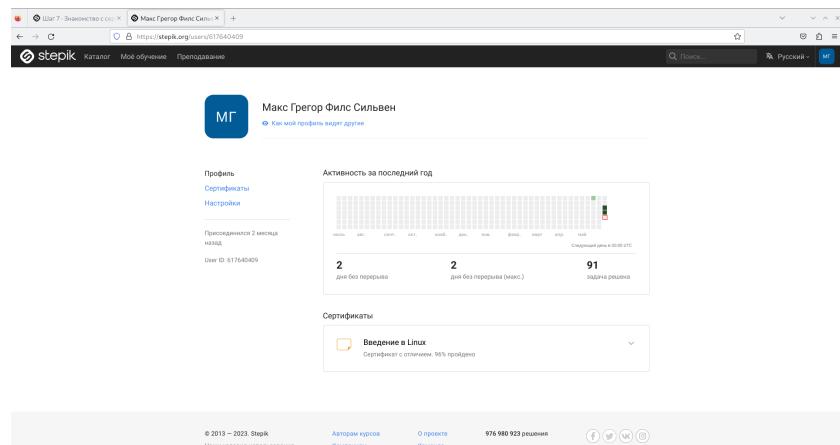


Рис. 3.1: Профиль в Stepik

1. Введение Вопрос: как называется этот курс? Чтобы ответить, выберите правильный ответ нажмите на зелёную кнопку ниже.
 - Здесь курс называется Введение в Linux, поэтому мы выбираем ответ : введение в Linux.(рис. 3.2).

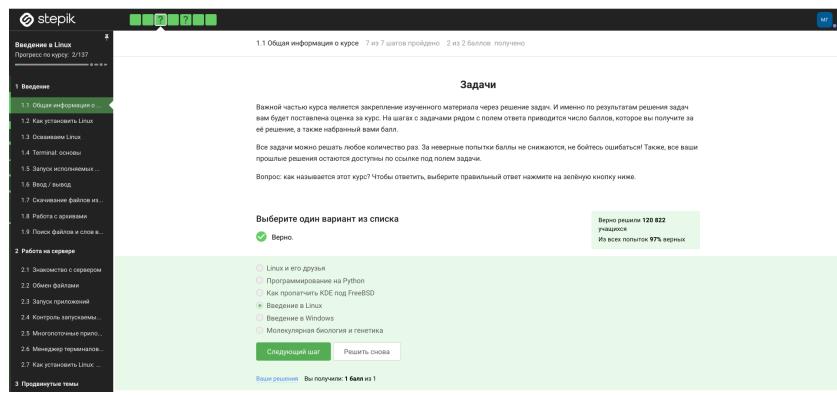


Рис. 3.2: Ответ на вопрос

2. Мы читаем текст и выбираем правильные ответы

(рис. 3.3)

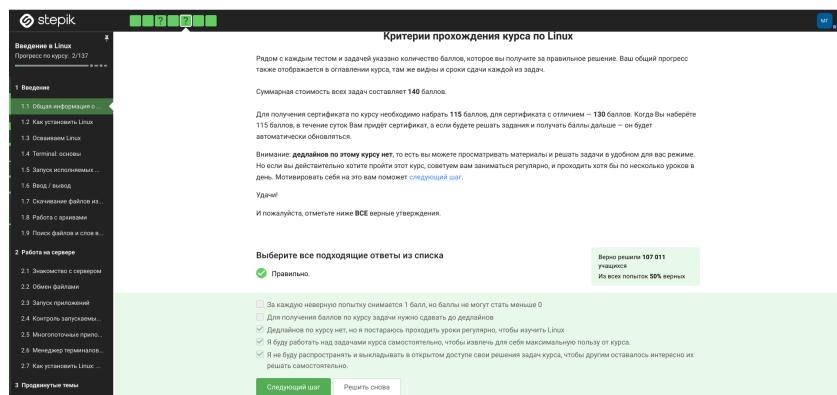


Рис. 3.3: Ответ на вопрос

3. Как установить Linux. Какую операционную систему вы обычно используете? В таких типах задания (с галочками/чекбоксами/checkbox) вы можете выбирать несколько вариантов ответа (от 0 до всех)!

- В настоящее время я использую Mac Os, поэтому я выбрал систему OS X (рис. 3.4).

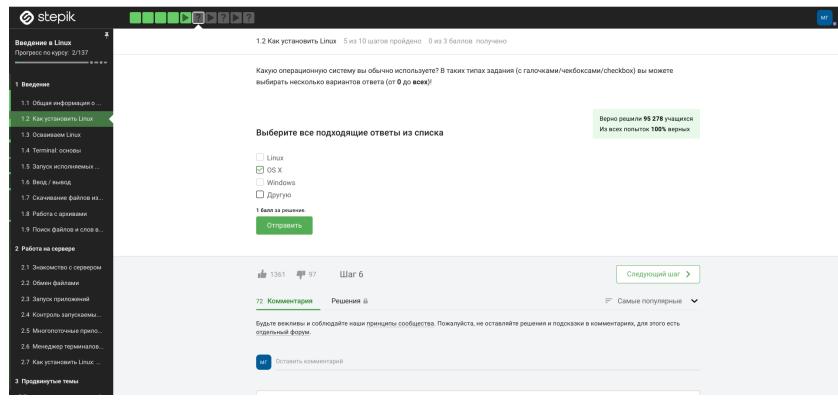


Рис. 3.4: Ответ на вопрос

4. Что такое виртуальная машина? Выберите наиболее подходящий ответ!
В таком типе заданий (с радиокнопками/radio button) ответ всегда ровно один!

- Виртуализация позволяет использовать операционные системы в таком приложении, как virtualbox(рис. 3.36).

Рис. 3.5: Ответ на вопрос

5. Смогли ли вы запустить на своем компьютере Linux?(рис. 3.6).

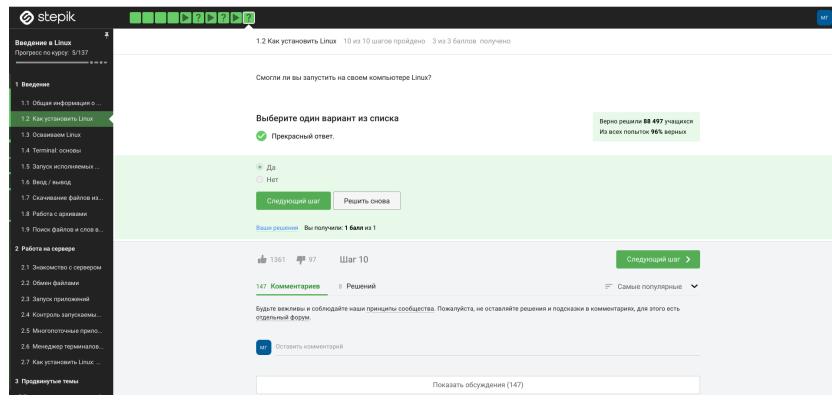


Рис. 3.6: Ответ на вопрос

6. Создайте документ в OpenOffice/LibreOffice Writer (аналог Microsoft Word) и напишите в нём шрифтом FreeMono (если такого шрифта у вас нет, то используйте Arial или Times New Roman) одну-единственную строчку: Hello, Linux! После этого сохраните этот документ в формате XML (Microsoft Word 2003 XML) или в формате FODT (OpenDocument Text: Flat XML) и загрузите в форму ниже.

- Мы только следили за этапами (рис. 3.7).

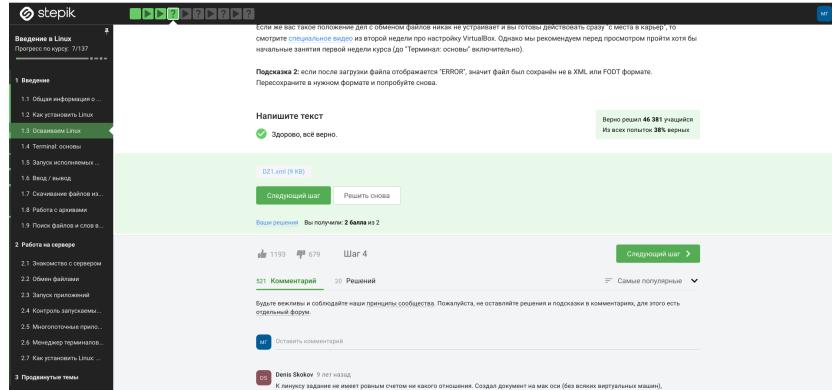


Рис. 3.7: Ответ на вопрос

7. Какое расширение имеют установочные пакеты в Linux (Ubuntu)?(рис. ??).

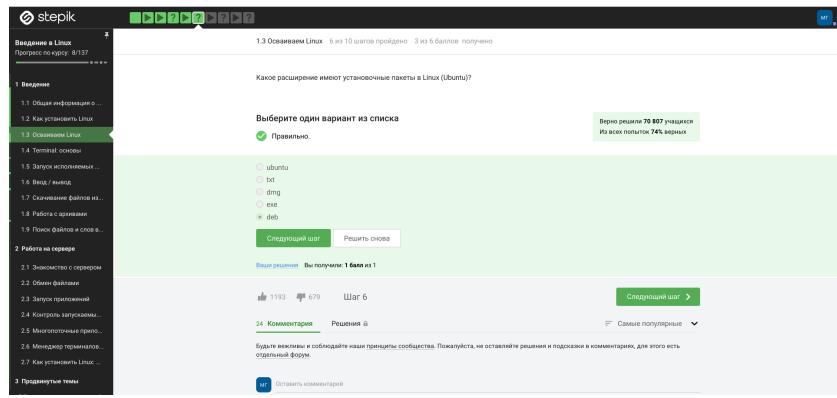


Рис. 3.8: Ответ на вопрос

- Поставьте себе в систему плеер VLC (любым способом: через Software Center или скачиванием установочного пакета с сайта VLC). Запустите, откройте Help About (или Shift+F1) и напишите ниже первую фамилию (без имени!) из вкладки Authors. Обратите внимание, что в англоязычных текстах обычно имя стоит на первом месте (first name), а фамилия на втором (last name).(рис. 3.9).

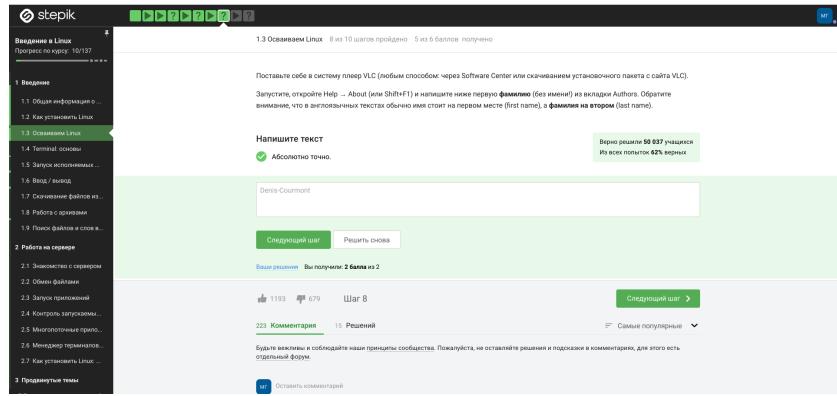


Рис. 3.9: Ответ на вопрос

- Для чего можно использовать приложение Update Manager?(рис. 3.10).

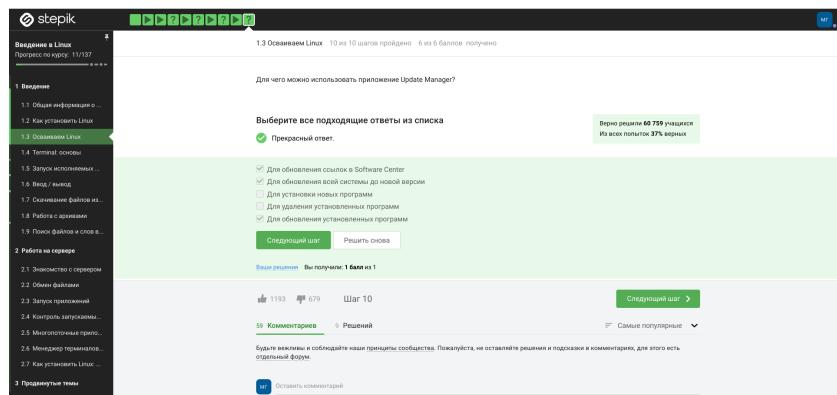


Рис. 3.10: Ответ на вопрос

10. Terminal: основы Выберите все синонимы для “командной строки”.(рис. 3.11).

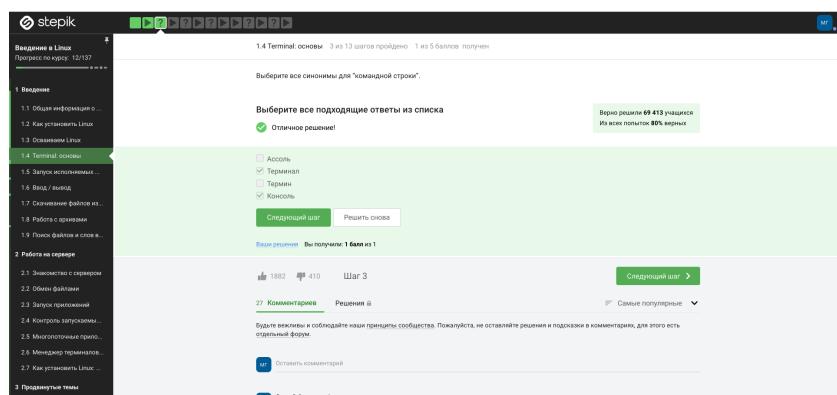


Рис. 3.11: Ответ на вопрос

11. Какая команда напечатает в какой директории мы сейчас находимся?(рис. 3.12).

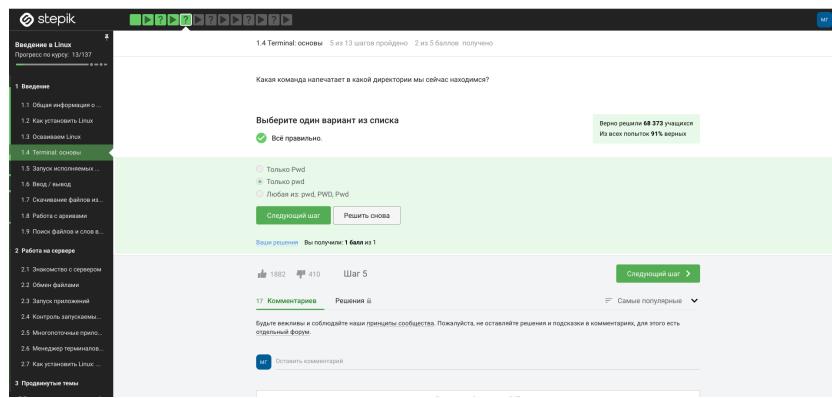


Рис. 3.12: Ответ на вопрос

12. Укажите, какие из следующих команд полностью эквивалентны команде `ls -A -human-readable -l /some/directory` (рис. 3.13).

Рис. 3.13: Ответ на вопрос

13. Предположим, что вы находитесь в директории `/home/bi/Documents`, причем `/home/bi` — ваша домашняя директория. Какая(ие) команда выведет содержимое `/home/bi/Downloads`, при этом не показывая содержимое других директорий?(рис. 3.14).

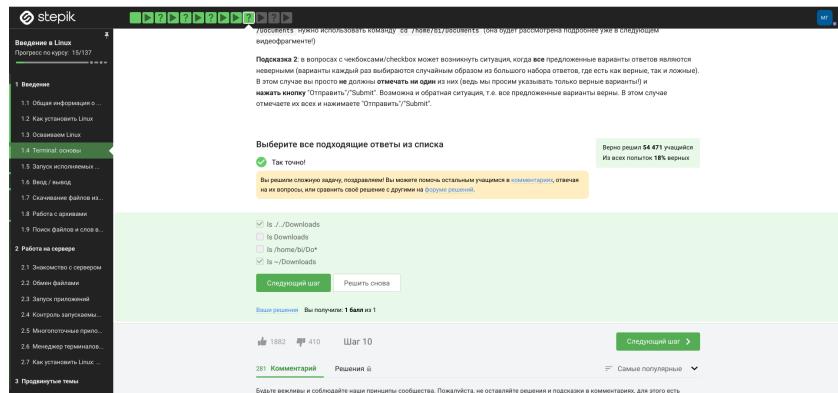


Рис. 3.14: Ответ на вопрос

14. Какая команда используется для удаления директорий?(рис. 3.15).

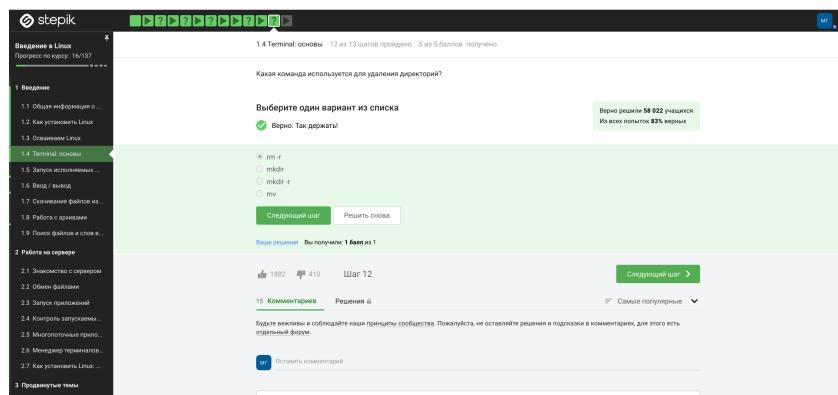


Рис. 3.15: Ответ на вопрос

15. Запуск исполняемых файлов Что произойдет, если ввести в терминал команду firefox (для запуска одно- именного браузера), а затем ввести туда же команду exit?(рис. 3.16).

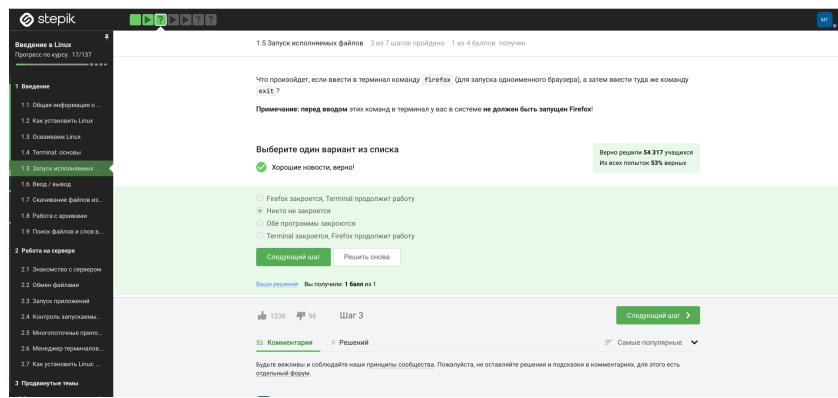


Рис. 3.16: Ответ на вопрос

16. Чему эквивалентен запуск программы с &?(рис. 3.17).

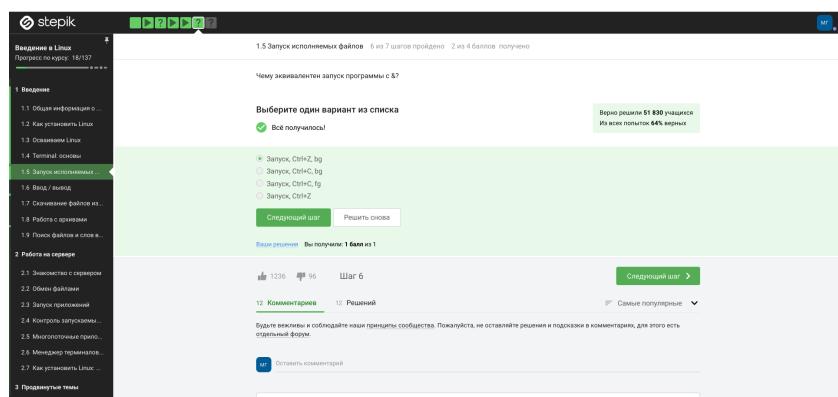


Рис. 3.17: Ответ на вопрос

17. Скачайте файл с программой, сделайте его исполняемым, запустите и скопируйте то, что он выведет на экран, в форму ниже.

- Мы поместили инструкции в онлайн-ридер python, и терминал этого компилятора выдал нам этот ответ.(рис. 3.18).

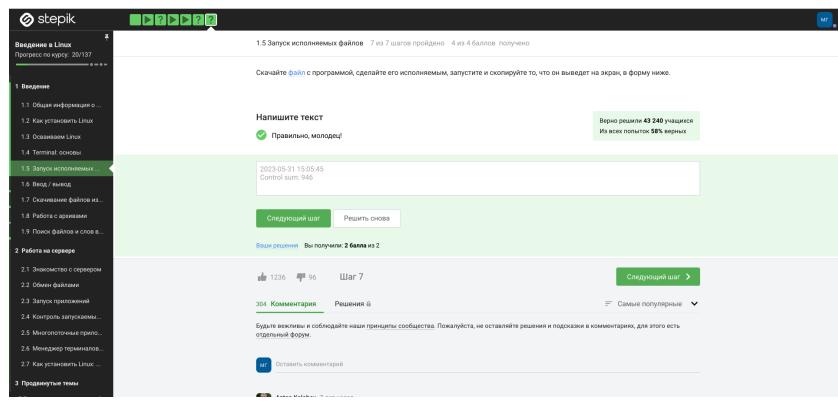


Рис. 3.18: Ответ на вопрос

18. Ввод / вывод Куда по умолчанию выводится поток ошибок из программы, запущенной в терминале?(рис. 3.19).

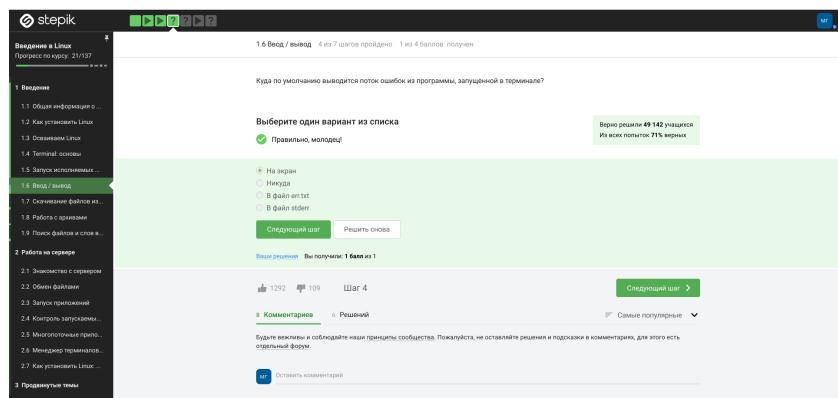


Рис. 3.19: Ответ на вопрос

19. Какие (какая) из команд создадут файл file.txt и запишут в него поток ошибок программы program? Считайте, что в момент запуска программы файл file.txt не существует(рис. 3.20).

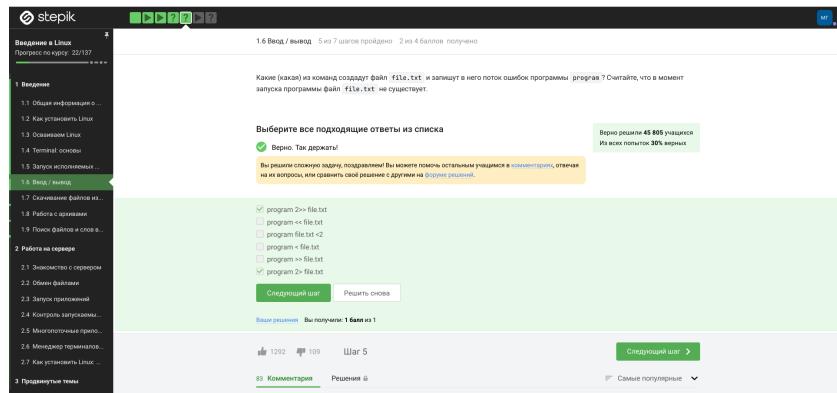


Рис. 3.20: Ответ на вопрос

20. Куда деваются сообщения об ошибках (т.е. вывод в stderr) от тех программ, которые объединены в конвейер (pipe)?(рис. 3.21).

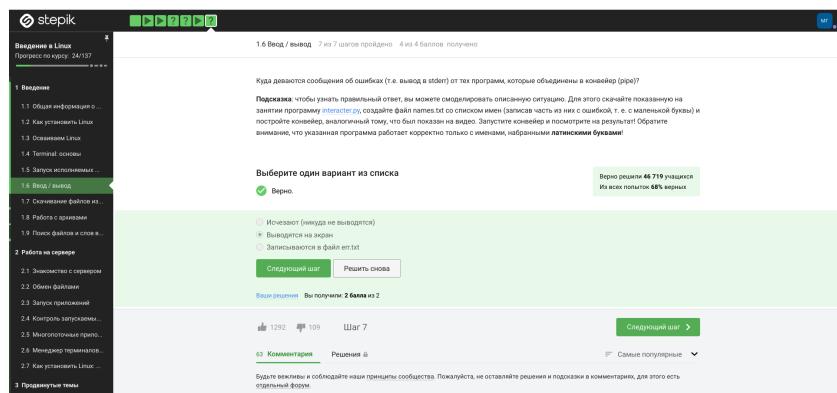


Рис. 3.21: Ответ на вопрос

21. Скачивание файлов из интернета В каком файле на диске окажется картинка, если для её скачивания были выполнены следующие команды? cd /home/alex/ wget -P /home/alex/Pictures -O 1.png http://example.com/example.png(рис. 3.22).

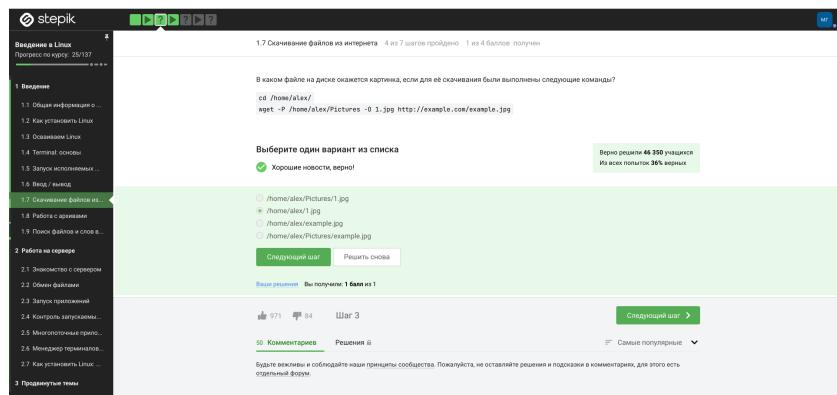


Рис. 3.22: Ответ на вопрос

22. Какую опцию нужно указать команде wget, чтобы она не выводила никаких сообщений на экран (Resolving.., Connecting to.. и т.д.)?

- Ни один из других ответов недействителен(рис. 3.23).

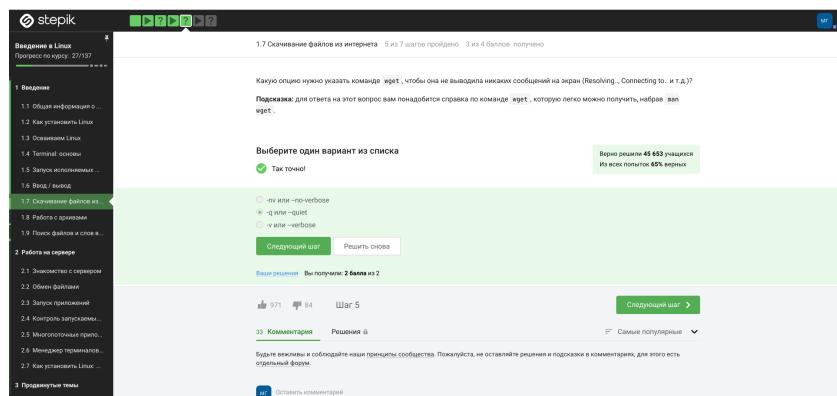


Рис. 3.23: Ответ на вопрос

23. Пусть на некоторой web-странице есть ссылки на картинки в форматах png и jpg, а также ссылки на другие страницы сайта (обычные html файлы). Какие файлы будут скачаны на компьютер, если запустить wget -r -l 1 -A png и передать в качестве аргумента ссылку на эту web-страницу? Выберите наиболее полный ответ! (рис. 3.24).

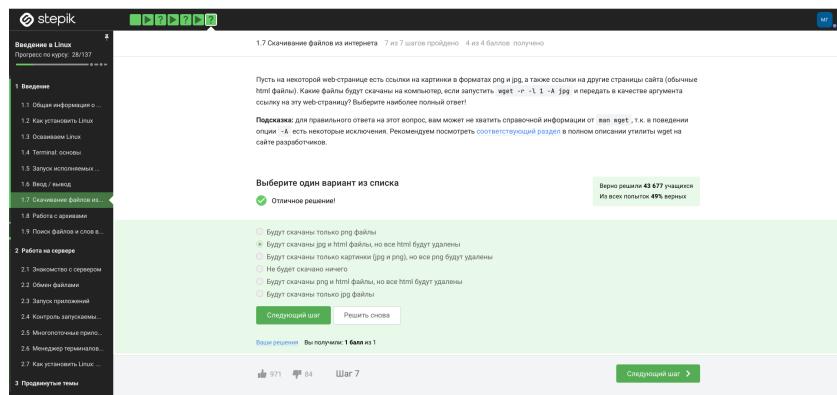


Рис. 3.24: Ответ на вопрос

24. Работа с архивами Чем отличаются архиваторы gzip и zip?

- Gzip работает быстрее, чем ZIP, при сжатии и распаковке. ZIP - это инструмент архивации и сжатия в одном флаконе, в то время как Gzip нуждается в помощи команды Tar для архивирования файлов. Gzip может сэкономить больше места на диске, чем приложения для сжатия ZIP-файлов. Как Gzip, так и ZIP используют алгоритм сжатия DEFLATE.(рис. 3.25).

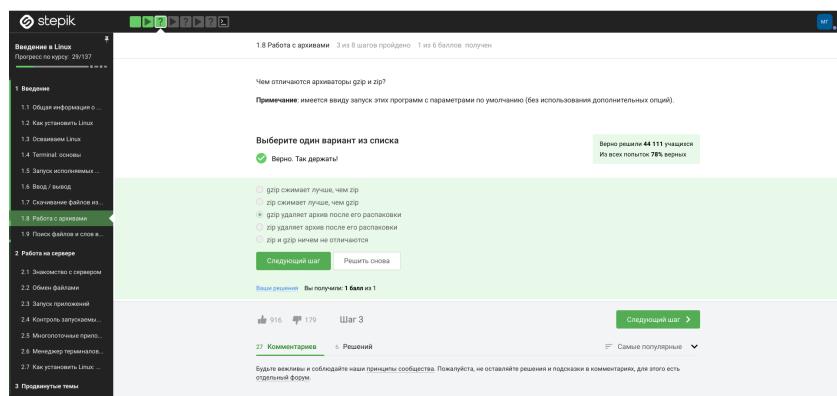


Рис. 3.25: Ответ на вопрос

25. Какие из перечисленных программ-архиваторов могут создать архив из директории с файлами?

- Оба являются компрессорами, а также позволяют архивировать документы.(рис. 3.26).

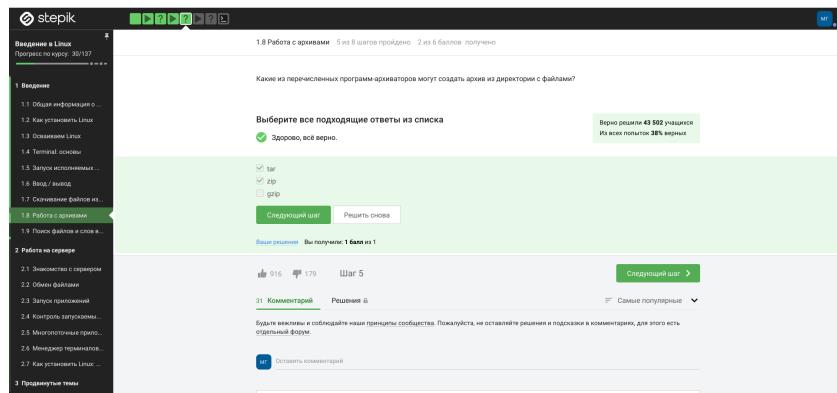


Рис. 3.26: Ответ на вопрос

26. Какой набор опций нужно указать программе tar, чтобы запаковать файлы в my_archive.tar.bz2?(рис. 3.27).

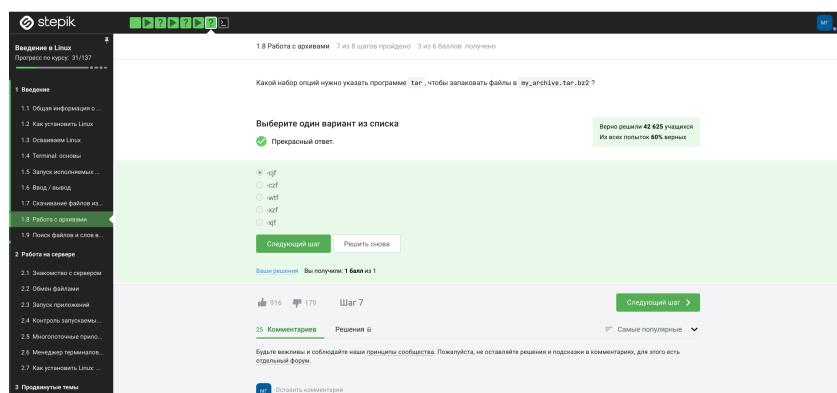


Рис. 3.27: Ответ на вопрос

27. Мы использовали терминал и загрузили папку с помощью команды wget, затем мы использовали команду tar для распаковки загруженного файла и получили следующий результат.(рис. 3.28).

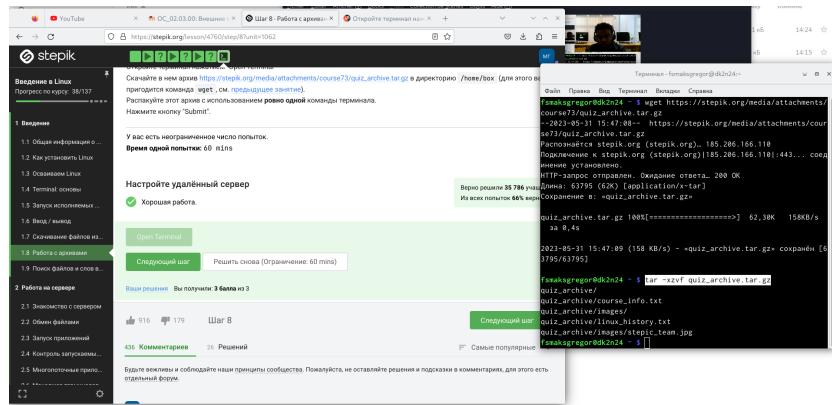


Рис. 3.28: Ответ на вопрос

28. Какая маска команды find НЕ найдет файл Alexey.jpeg?(рис. 3.29).

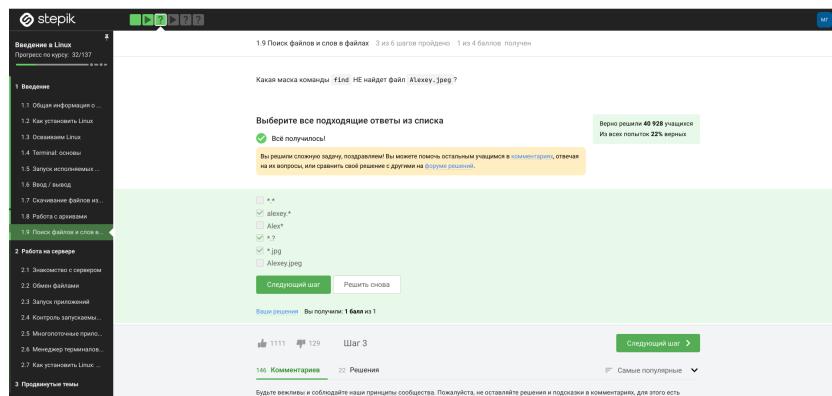


Рис. 3.29: Ответ на вопрос

29. Предположим, что в файле text.txt записаны строки, показанные среди вариантов ответа. Отметьте только те из них, которые выведет на экран команда grep "world" text.txt.

- мы использовали терминал нашего компьютера, вводили разные тексты и отмечали правильные ответы(рис. 3.30).

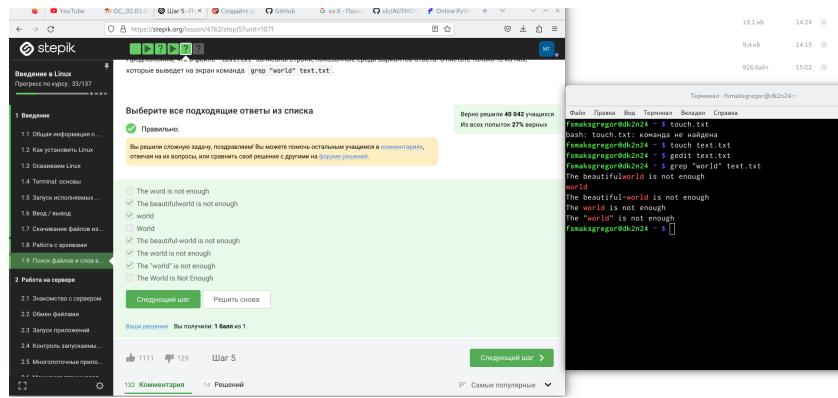


Рис. 3.30: Ответ на вопрос

30. Скачайте архив с произведениями Шекспира. Вам нужно сгенерировать файл, в котором будут все строчки из этих произведений, содержащие “love”, и загрузить этот файл в форму.

- Мы внесли необходимую информацию в файл outlog.txt затем загрузите на сайт(рис. 3.31).

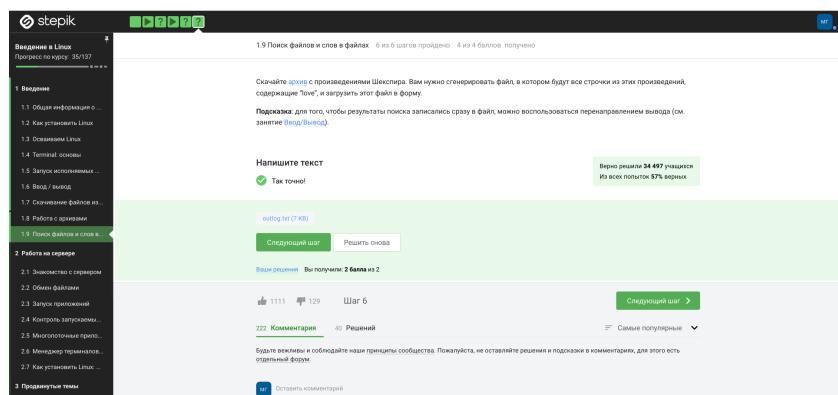


Рис. 3.31: Ответ на вопрос

3.2 Работа на сервере

- Для каких задач можно использовать удаленный сервер?(рис. 3.32).

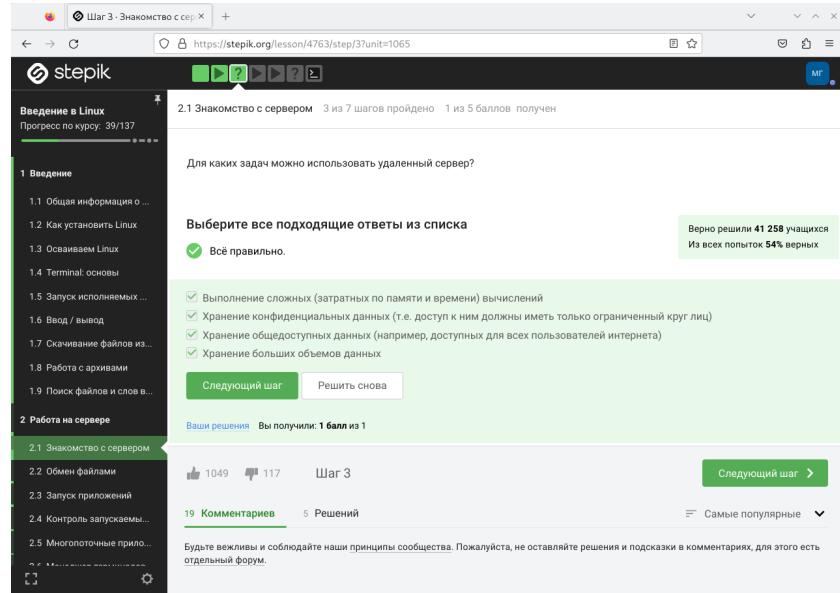


Рис. 3.32: Ответ на вопрос

2. Предположим программа ssh-keygen создала вам два ключа: `id_rsa` и `id_rsa.pub`. Какой из этих ключей можно без опаски пересыпать по интернету?(рис. 3.33).

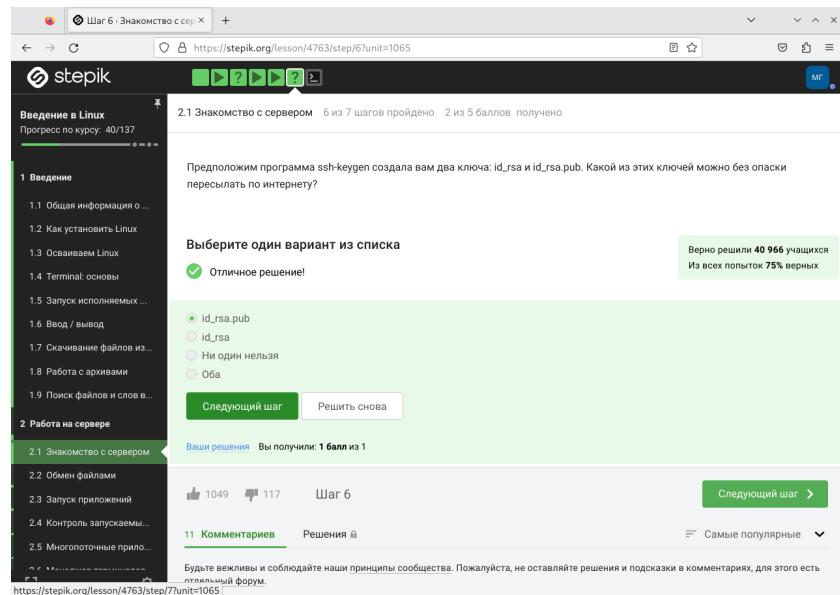


Рис. 3.33: Ответ на вопрос

3. Какая команда скопирует на сервер (в домашнюю директорию) папку stepic вместе с содержимым ее самой и всех ее подпапок?(рис. ??).

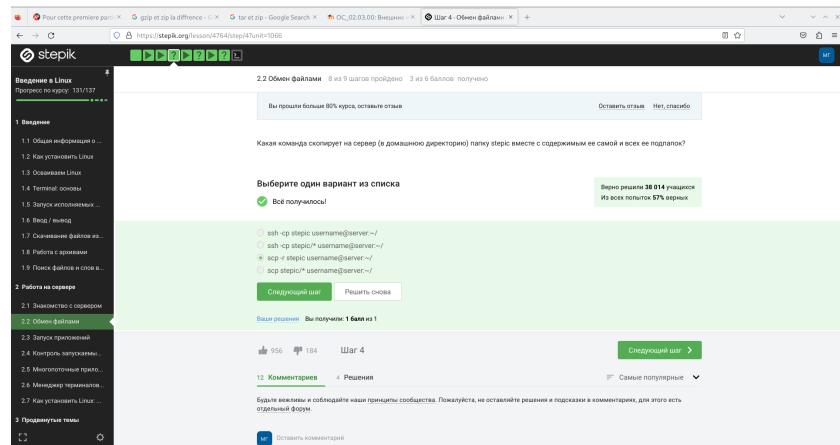


Рис. 3.34: Ответ на вопрос

4. Предположим, что вы устанавливаете программу program на свой компьютер при помощи команды sudo apt-get install program. Терминал сообщает вам, что он не может найти и скачать установочный пакет. Какие действия могут устранить проблему?(рис. 3.35).

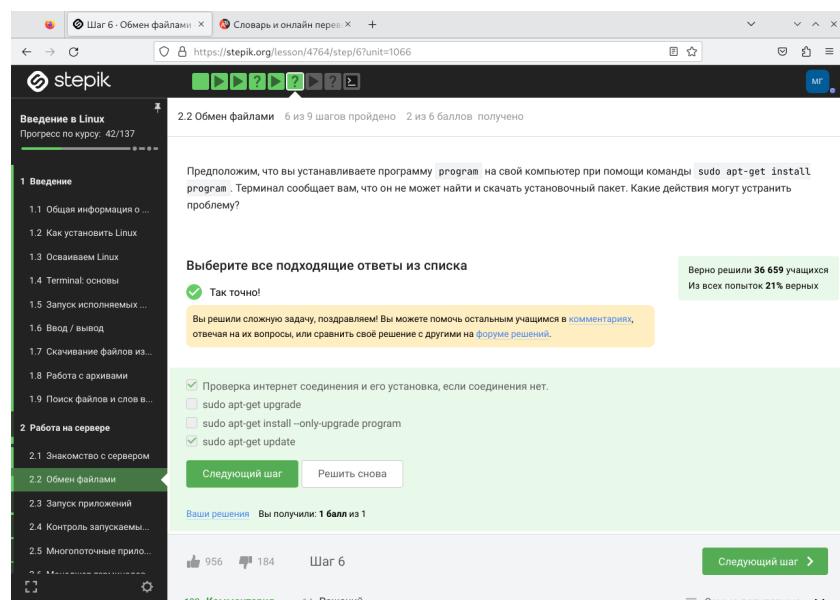


Рис. 3.35: Ответ на вопрос

5. Для чего можно использовать программу Filezilla? (рис. 3.36).

The screenshot shows a Stepik lesson page for "Введение в Linux". The current step is "2.2 Обмен файлами" (File Transfer) with 8 steps completed. The question asks: "Для чего можно использовать программу Filezilla?". Below it, a list of options is shown, with the first one checked: "✓ Для просмотра содержимого директорий на сервере". A green box indicates "Всё правильно." (All correct). At the bottom, there are buttons for "Следующий шаг" (Next step) and "Решить снова" (Solve again). The sidebar shows other steps like "2.1 знакомство с сервером" and "2.3 запуск приложений".

Рис. 3.36: Ответ на вопрос

6. Что можно сделать, если требуется запустить на сервере программу, для работы которой нужен не терминал, а экран?(рис. 3.37).

The screenshot shows a Stepik lesson page for "Введение в Linux". The current step is "2.3 Запуск приложений" with 4 steps completed. The question asks: "Что можно сделать, если требуется запустить на сервере программу, для работы которой нужен не терминал, а экран?". Below it, a list of options is shown, with the first one checked: "✓ Проверить, есть ли другая версия этой программы (специально для терминала)". A green box indicates "Верно решили 35 473 учащихся" (Correctly solved by 35 473 students). At the bottom, there are buttons for "Следующий шаг" (Next step) and "Решить снова" (Solve again). The sidebar shows other steps like "2.1 знакомство с сервером" and "2.4 контроль запускаемыми...".

Рис. 3.37: Ответ на вопрос

7. Как обычно можно вызвать справочную информацию о программе program(рис. 3.38).

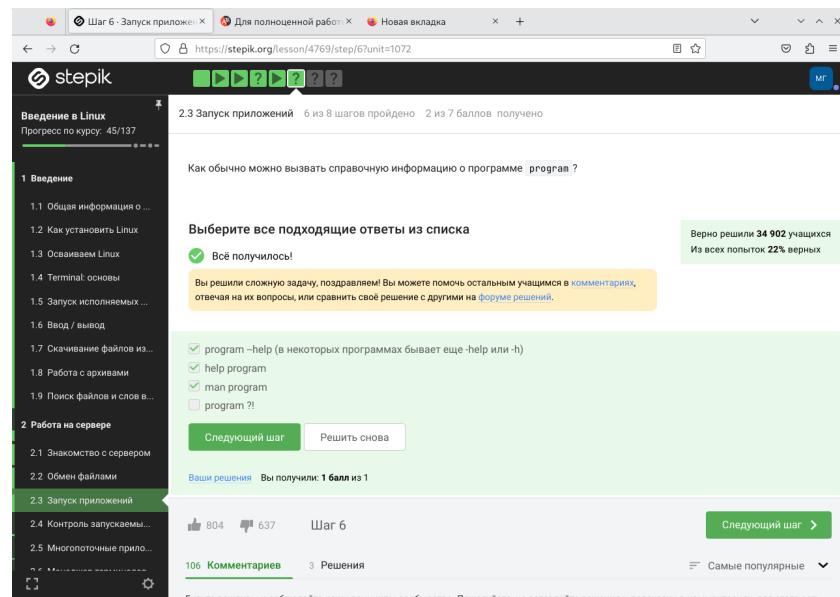


Рис. 3.38: Ответ на вопрос

8. Посмотрите справку по программе FastQC (имеется ввиду вариант для запуска в терминале) и определите, какие форматы данных он может принимать на вход. Если вы хотите попробовать запустить FastQC на каких-то реальных данных, то можете попробовать на этом файле.(рис. 3.39).

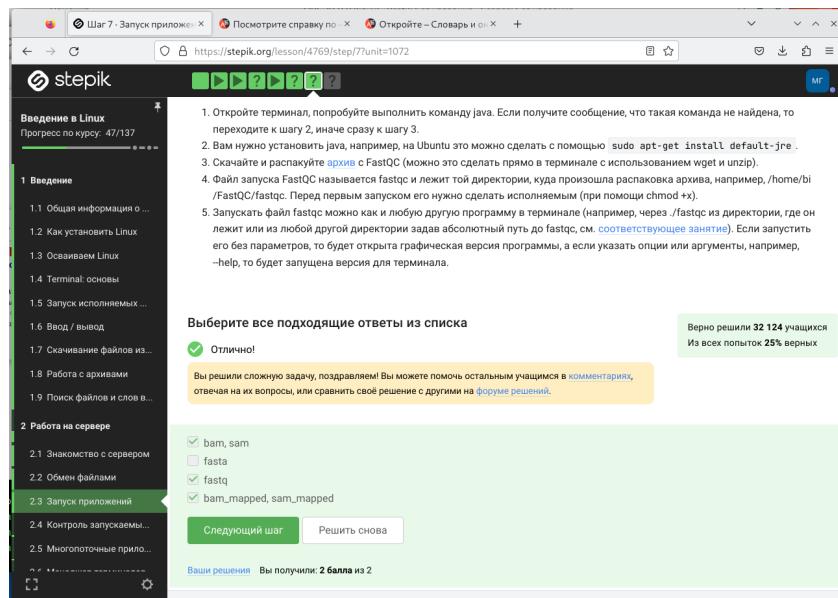


Рис. 3.39: Ответ на вопрос

9. Посмотрите справку по программе (имеется в виде версия для терминала) и впишите в поле ниже команду, которая запускает в терминале Clustal на файле test.fasta и выполняет множественное выравнивание (multiple alignment). Никакие лишние опции указывать не нужно (только необходимые для выполнения этого задания)!(рис. 3.40).

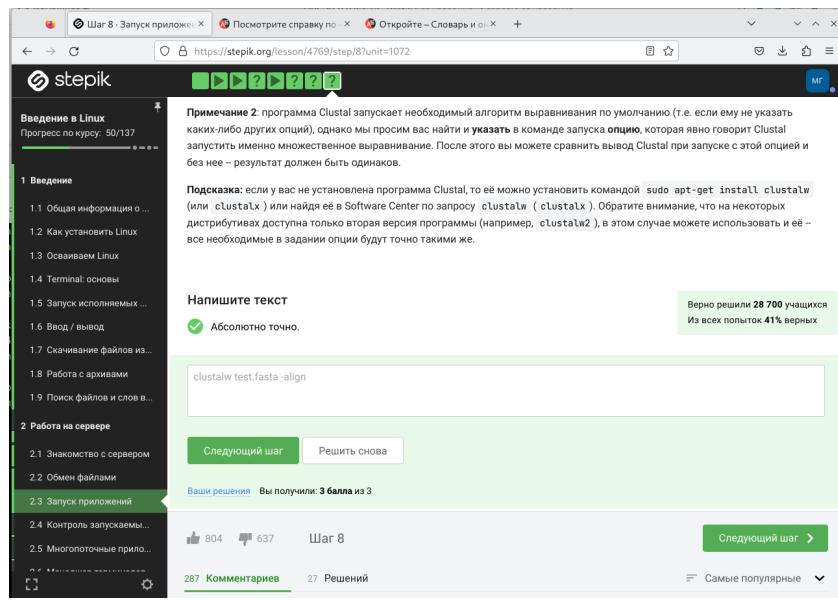


Рис. 3.40: Ответ на вопрос

10. Контроль запускаемых программ Предположим вы запустили программы program1, program2 и program3 в фоновом режиме. После этого вы выполнили следующие действия: fg %1 Ctrl+C fg%2 Ctrl+Z jobs Информация о каких программах будет показана при выполнении команды jobs(рис. 3.41).

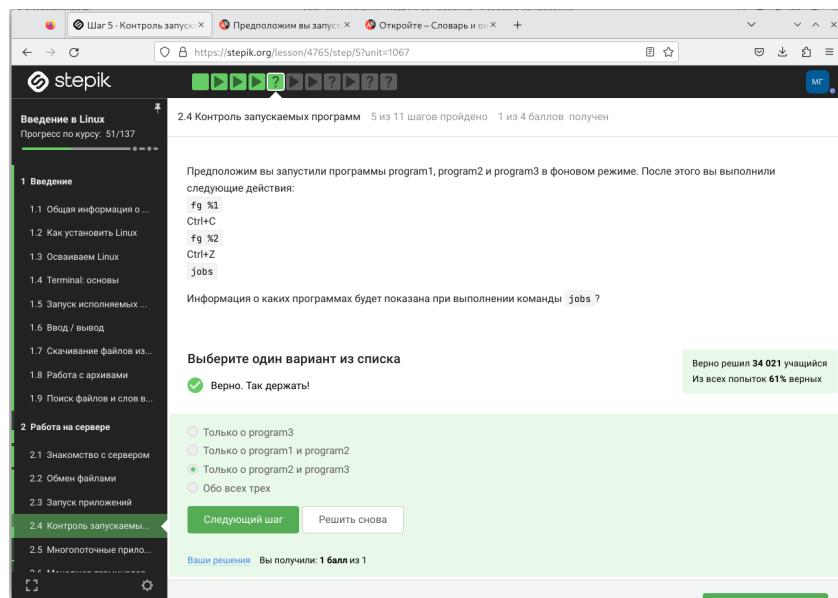


Рис. 3.41: Ответ на вопрос

11. В каждой из этих трех утилит для каждой запущенной программы указывается число-идентификатор. Одноковые ли эти идентификаторы в jobs, top и ps? (рис. 3.42).

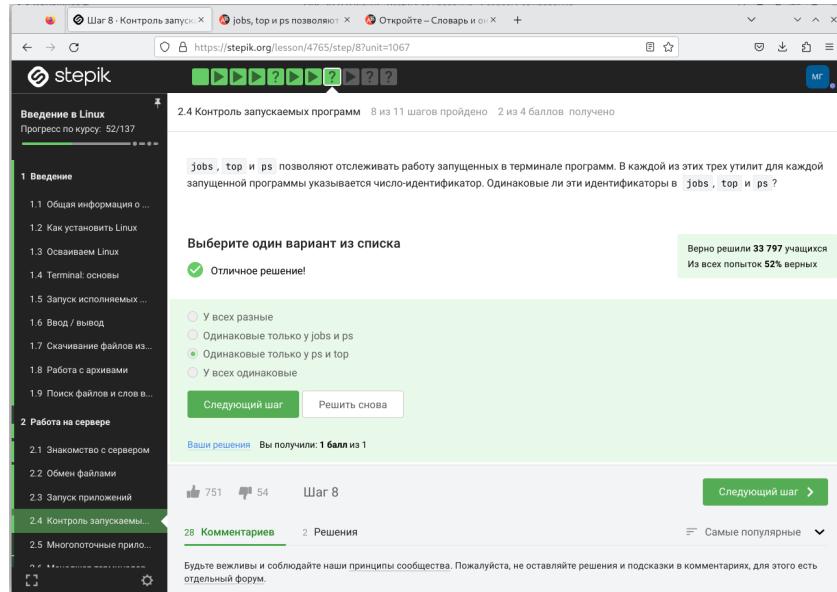


Рис. 3.42: Ответ на вопрос

12. С помощью какой команды можно мгновенно завершить остановленный про-цесс?(рис. ??).

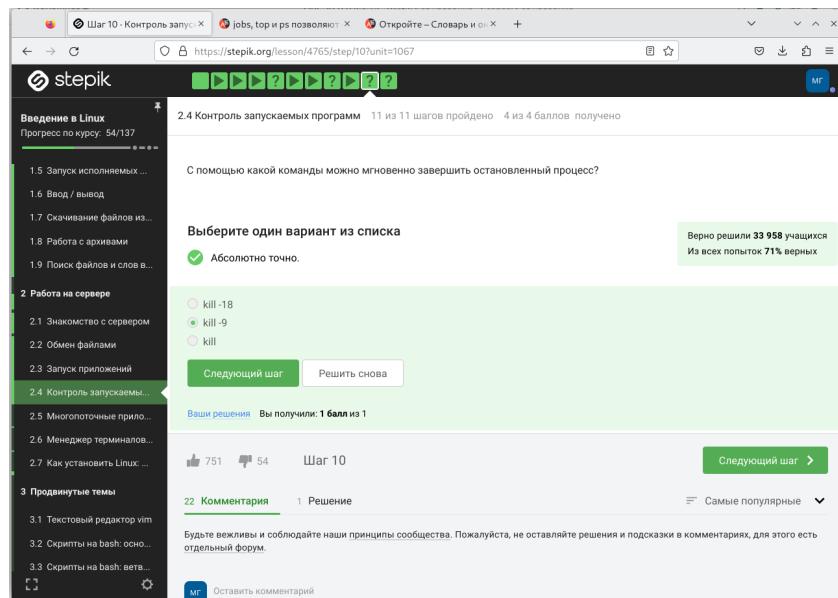


Рис. 3.43: Ответ на вопрос

13. Что произойдет, если использовать `kill` (без опций) по отношению к процессу, который был приостановлен при помощи `Ctrl+Z?` (рис. 3.44).

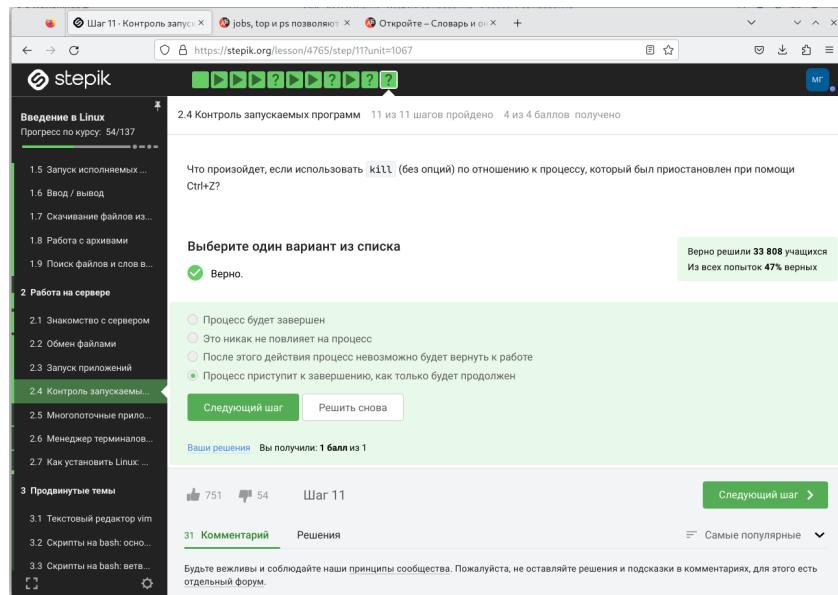


Рис. 3.44: Ответ на вопрос

14. Многопоточные приложения Сколько вычислительных ресурсов централь-

ного процессора (% CPU) использует остановленное (по Ctrl+Z) многопоточное приложение? Учитывайте, что 100% CPU означает загрузку одного процессора, 200% CPU – двух процессоров (на многопроцессорных и/или многоядерных компьютерах) и т.д. Например, выполняющееся в 4 потока приложение обычно использует около 400% CPU, однако наш вопрос касается именно момента после остановки такого приложения.(рис. 3.45).

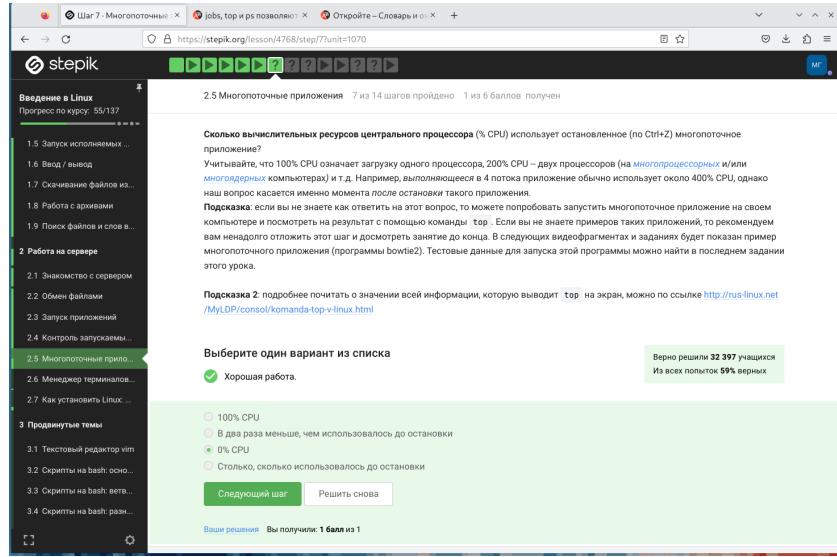


Рис. 3.45: Ответ на вопрос

15. Сколько памяти занимает остановленное (по Ctrl+Z) многопоточное приложение? Подсказка: если вы не знаете как ответить на этот вопрос, то можете попробовать запустить многопоточное приложение на своем компьютере и посмотреть на результат с помощью команды `top`. Если вы не знаете примеров таких приложений, то рекомендуем вам ненадолго отложить этот шаг и досмотреть занятие до конца. В следующих видеофрагментах и заданиях будет показан пример многопоточного приложения (программы `bowtie2`). Тестовые данные для запуска этой программы можно найти в последнем задании этого урока.(рис. 3.46).

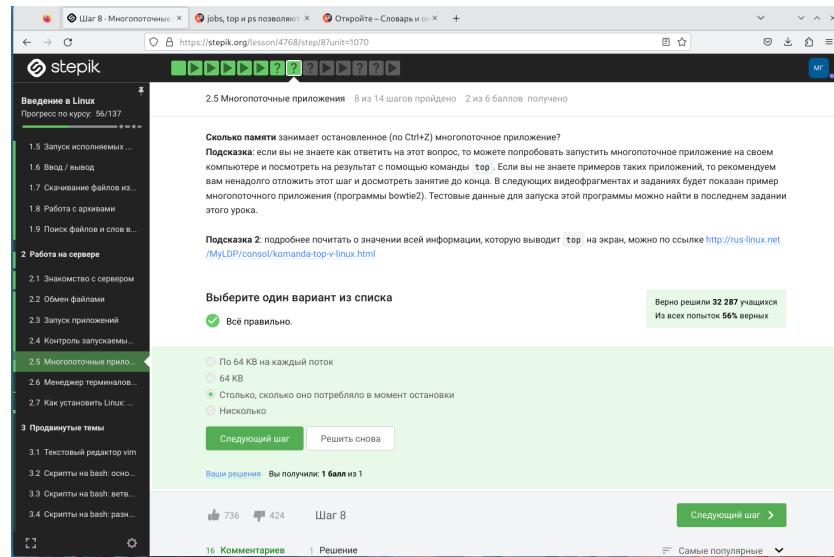


Рис. 3.46: Ответ на вопрос

15. Как принудительно завершить один из потоков запущенного многопоточного приложения?(рис. 3.47).

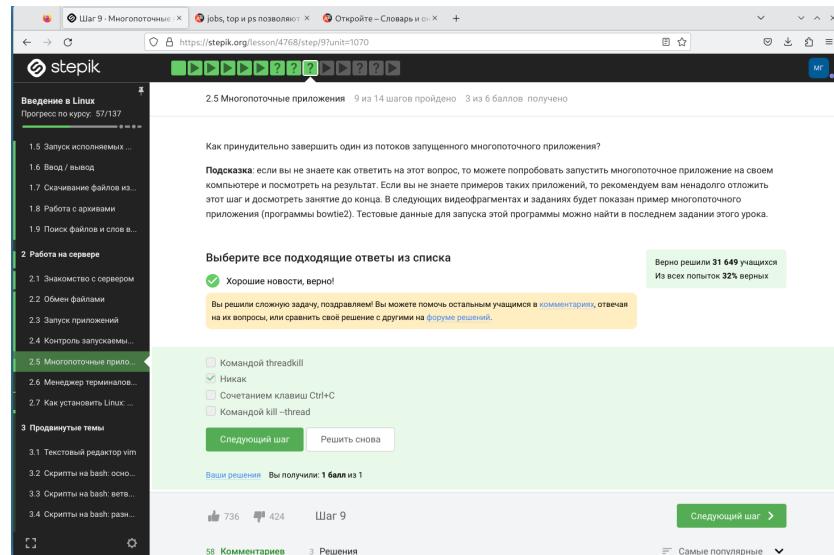


Рис. 3.47: Ответ на вопрос

16. Для выполнения этого задания вам потребуется программа bowtie2. Надеемся, что вы разобрались, что запуск bowtie2 состоит из двух шагов –сначала

запускаем подпрограмму bowtie2-build, а затем подпрограмму bowtie2. Изучите справочную информацию об этих подпрограммах (можно вызвать при помощи `-help`) и ответьте на вопрос какой(ие) из этих шагов можно выполнить в несколько потоков? (рис. 3.48).

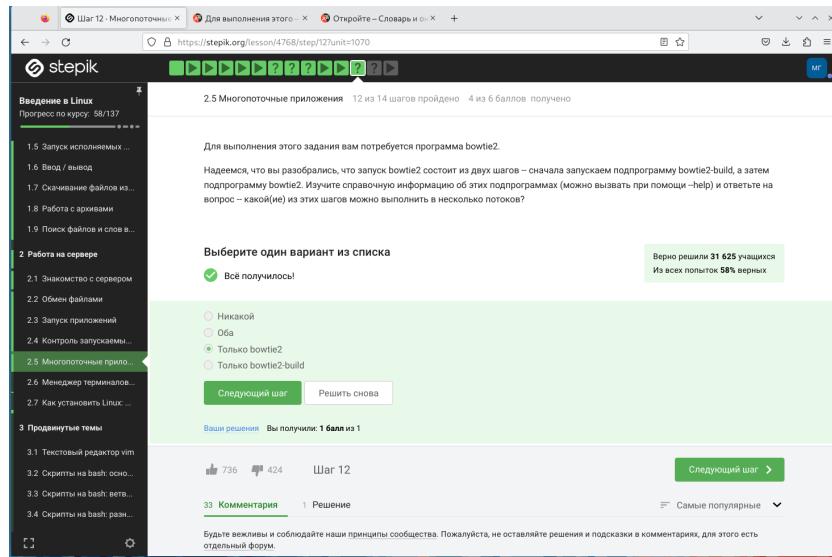


Рис. 3.48: Ответ на вопрос

17. Скачайте файлы, необходимые для запуска bowtie2: референсный геном (reference) и риды (reads). Запустите программу bowtie2 на этих данных (напоминаем, что запуск состоит из двух этапов!). Вывод stderr второго этапа (т.е. запуск подпрограммы bowtie2) запишите в файл (см. занятие про перенаправление ввода/вывода) и загрузите его в форму ниже. Мы также рекомендуем вам перенаправлять вывод stdout в файлы на обоих этапах, чтобы он не засорял экран вашего терминала. Попробуйте теперь запустить второй этап (запуск подпрограммы bowtie2) в несколько потоков. Рекомендуем выставить число потоков равное количествуядер на вашем компьютере (команда `proc`). Сравните скорость выполнения в таком режиме с работой в один поток. Также рекомендуем убедиться, что результаты запусков (т.е. вывод в stderr) полностью совпали в обоих режимах! (рис. 3.49).

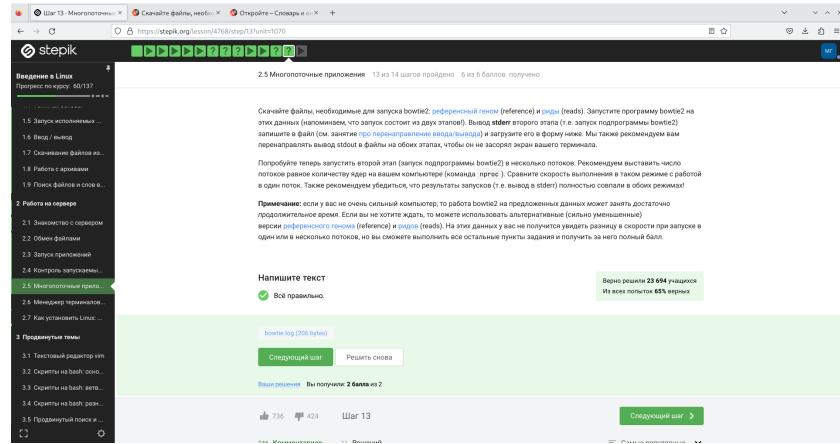


Рис. 3.49: Ответ на вопрос

18. Менеджер терминалов tmux Вы открыли две вкладки в терминале. В одной из них вы запустили процесс и приостановили его. Переключившись во вторую вкладку и набрав fg, вы добьетесь следующего:(рис. 3.50).

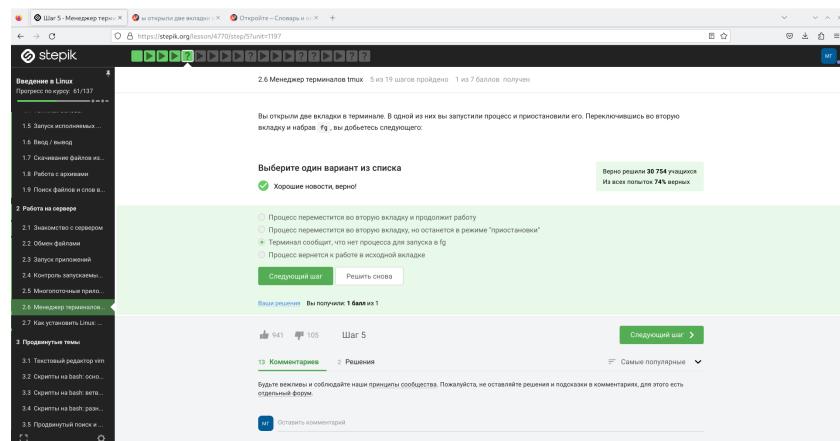


Рис. 3.50: Ответ на вопрос

19. Предположим, что в tmux осталась последняя открытая вкладка. Что произойдет, если вы введете в этой вкладке в командную строку команду exit?(рис. 3.52).

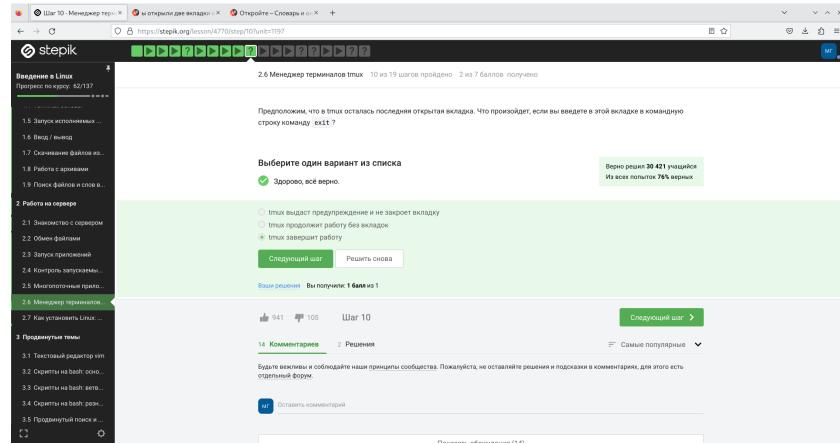


Рис. 3.51: Ответ на вопрос

20. Предположим, что вы открыли терминал, зашли в нем на сервер, запустили на этом сервере tmux и начали работу в нем. Что произойдет, если вы теперь закроете терминал?(рис. 3.52).

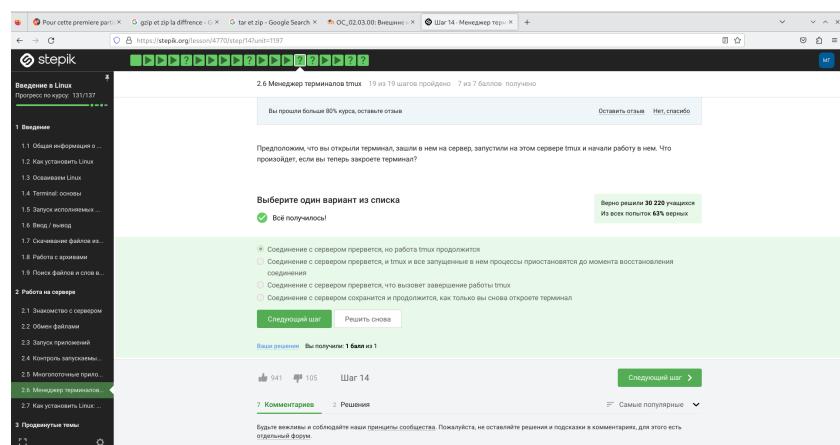


Рис. 3.52: Ответ на вопрос

21. Что произойдет, если запустить процесс в фоновом режиме в одной из вкладок tmux, а затем принудительно закрыть эту вкладку (Ctrl+B, X)?(рис. 3.53).

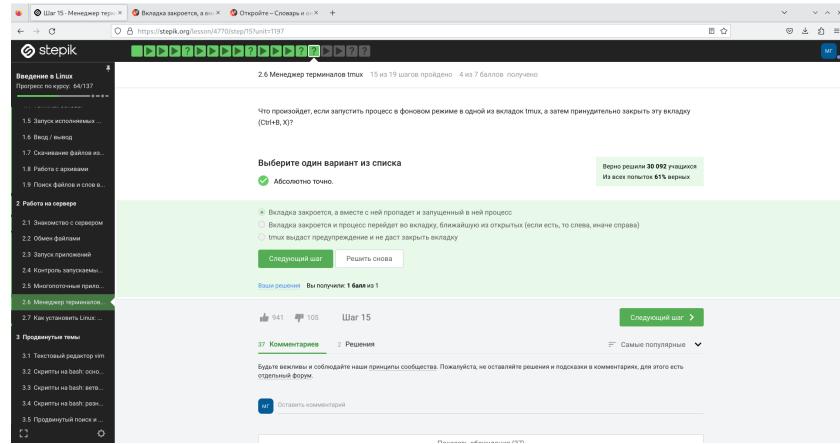


Рис. 3.53: Ответ на вопрос

22. Изучите справку по tmux (например, man tmux) и выберите из предложенных ниже tmux-команд ту, которая отвечает за переименование текущей вкладки.(рис. 3.54).

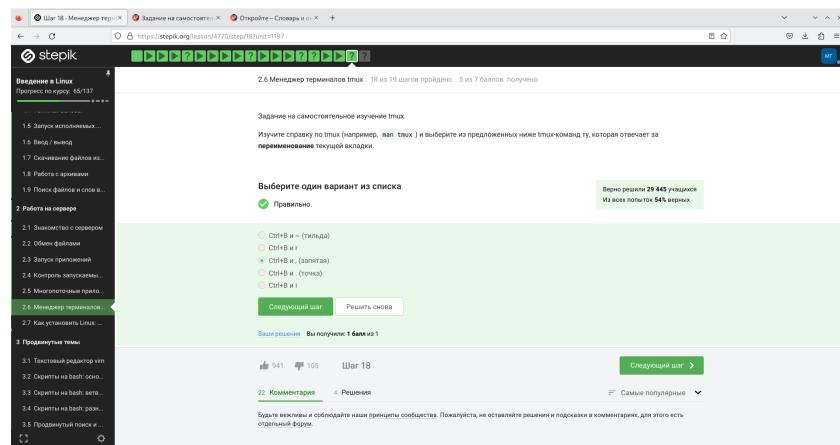


Рис. 3.54: Ответ на вопрос

23. Кроме создания нескольких вкладок, tmux умеет еще и разделять (split) одну вкладку на несколько, например, горизонтальной чертой на верхнюю и нижнюю или вертикальной чертой на левую и правую. Разделение может быть полезно, например, чтобы запустить процесс в верхней половине вкладки, а продолжить работу в нижней и одновременно следить за тем,

что происходит с процессом. Для “горизонтального” разделения используется (Ctrl+B и “), а для “вертикального” –(Ctrl+B и %). Предлагаем вам самостоятельное изучить работу с “вкладками внутри вкладок” и отметить верные утверждения из списка ниже. Вы можете использовать справку по tmux (например, man tmux) или просто попробовать воспроизвести эти утверждения у себя на компьютере.(рис. 3.84).

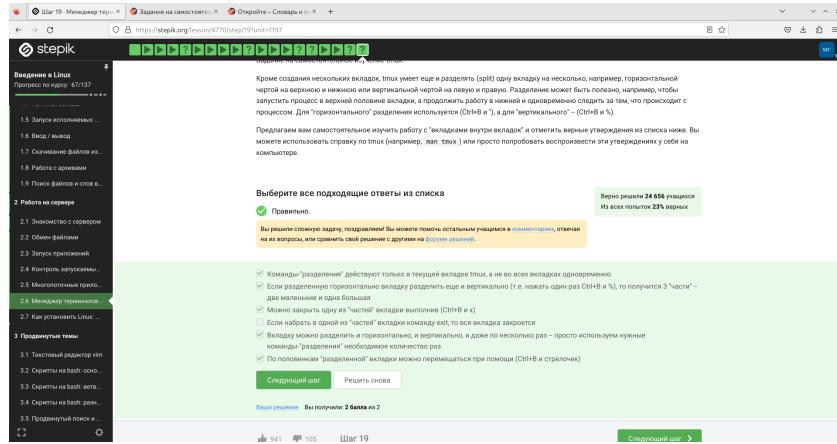


Рис. 3.55: Ответ на вопрос

3.3 Продвинутые темы

1. Текстовый редактор vim 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.(рис. 3.56).

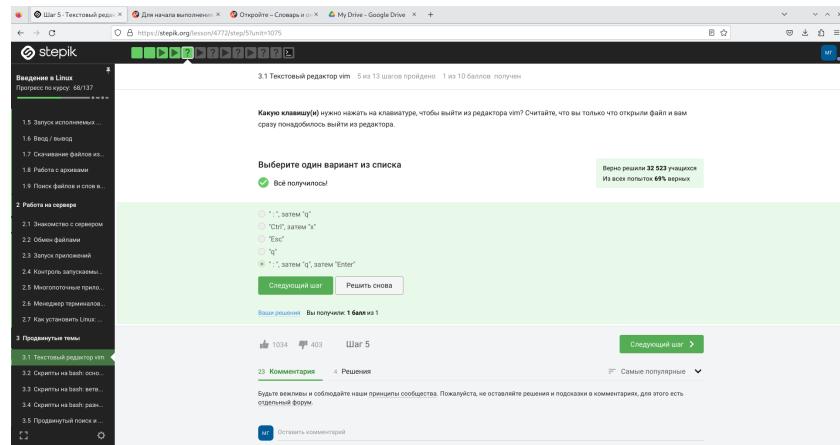


Рис. 3.56: Ответ на вопрос

2. При перемещении в vim “по словам” есть небольшая разница в том, используем мы маленькую (`w`, `e`, `b`) или большую (`W`, `E`, `B`) букву. Первые перемещают нас по “словам” (word), а вторые по “большим словам” (WORD). Посмотрите справку по этим перемещениям и разберитесь в чем заключается разница между `word` и `WORD`. А для того, чтобы убедиться, что вы разобрались, отметьте ниже все верные утверждения про следующую строку: `Strange_TEXT is_here. 2=2 YES!`(рис. 3.57).

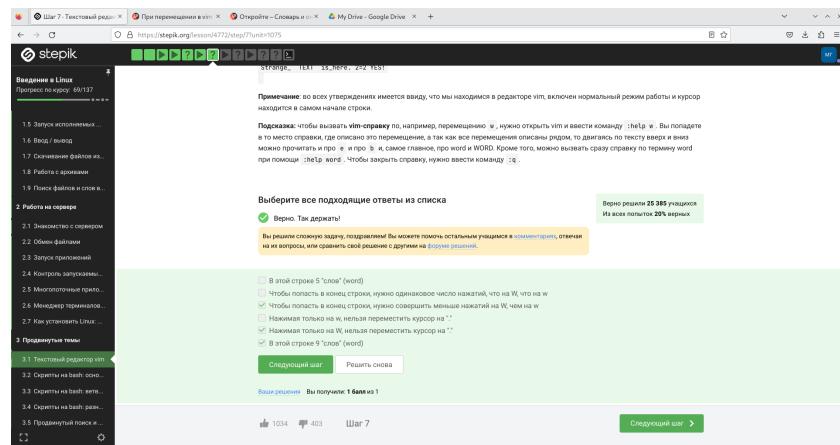


Рис. 3.57: Ответ на вопрос

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 (т.е. знаки “<” и “>” не несут отдельного смысла).(рис. 3.58).

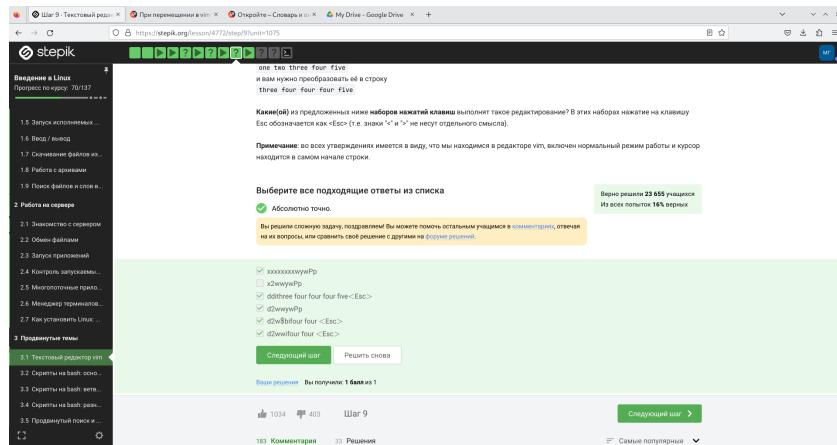


Рис. 3.58: Ответ на вопрос

- 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 42ввода команды обозначать никак не нужно.(рис. 3.59).

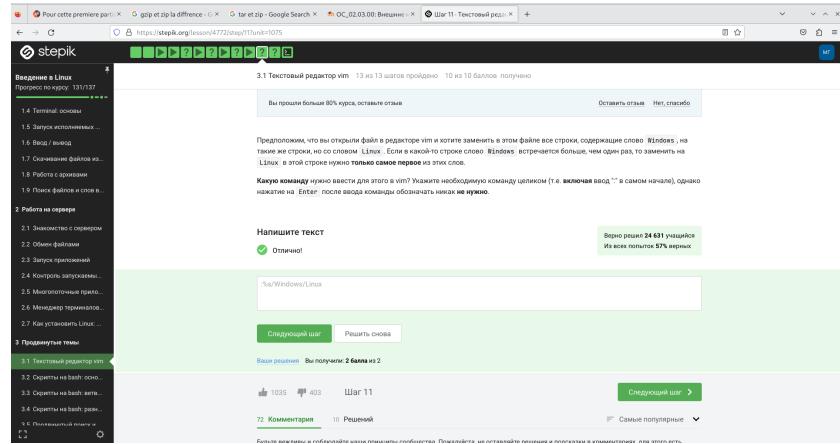


Рис. 3.59: Ответ на вопрос

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.(рис. 3.60).

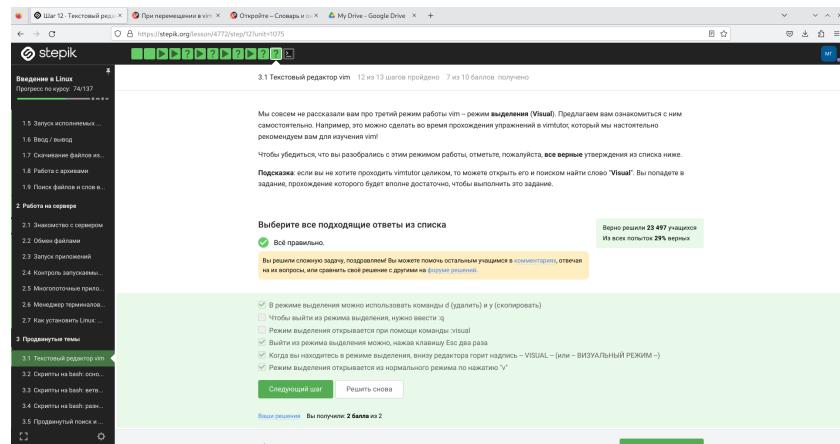


Рис. 3.60: Ответ на вопрос

6.

(рис. 3.61).

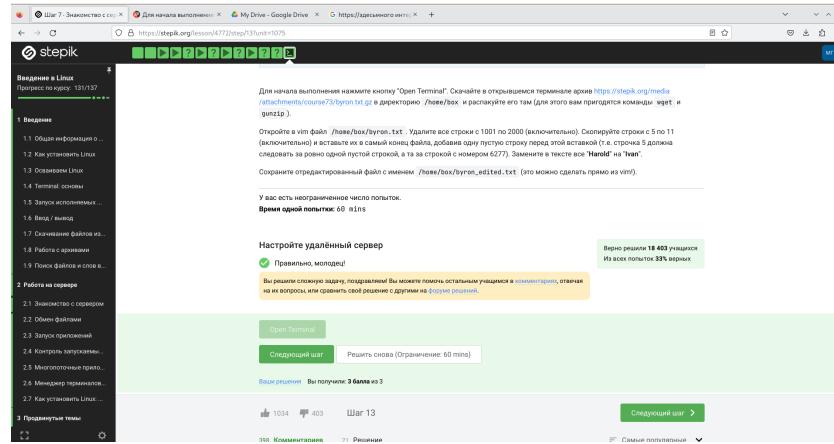


Рис. 3.61: Ответ на вопрос

7. Надеемся, что вы разобрались, что одну оболочку (например, sh) можно запускать 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?(рис. ??).

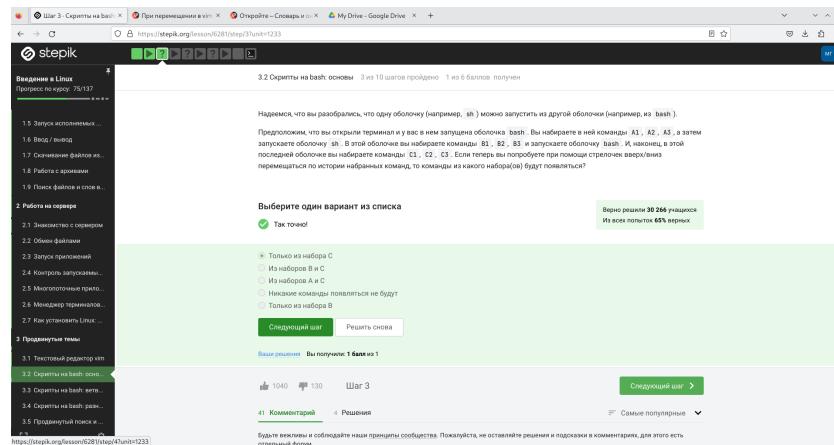


Рис. 3.62: Ответ на вопрос

8. 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 окончанию работы скрипта?(рис. 3.63).

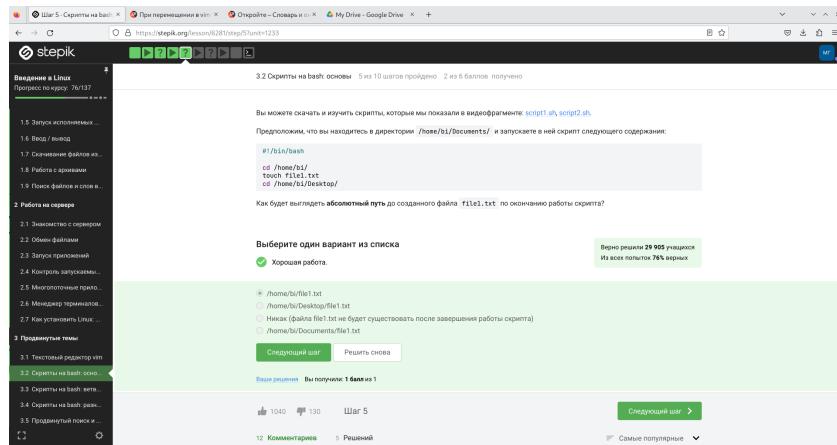


Рис. 3.63: Ответ на вопрос

9. 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”.(рис. 3.64).

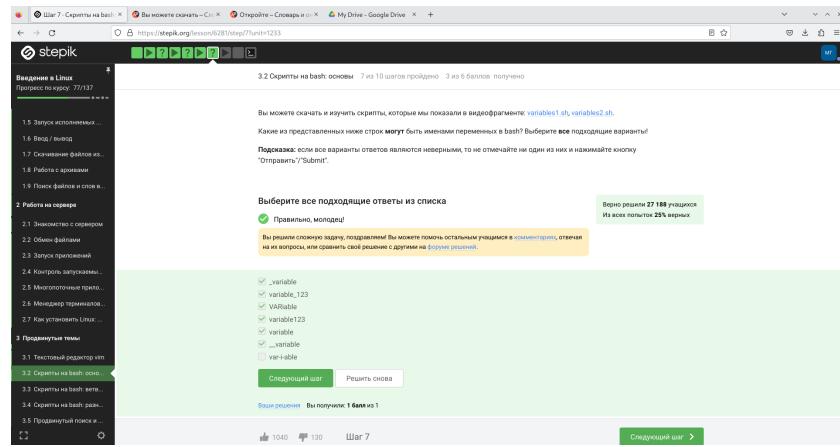


Рис. 3.64: Ответ на вопрос

10. 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(рис. 3.65).

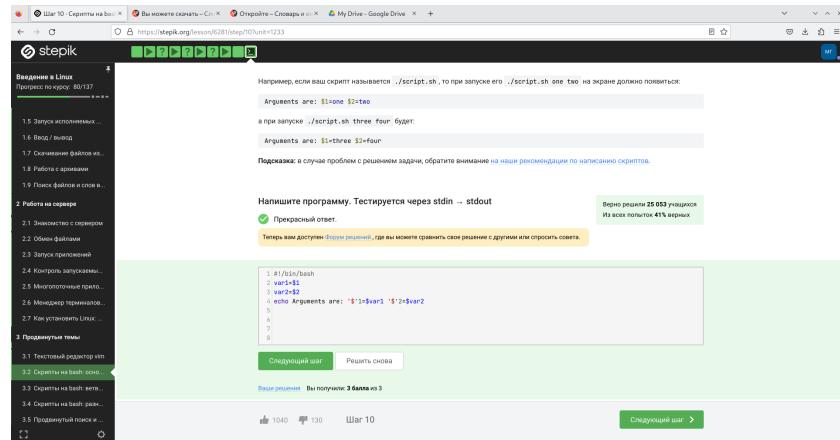


Рис. 3.65: Ответ на вопрос

11. Вы можете скачать и изучить скрипт, который мы показали в видеофрагменте: branching1.sh. Предположим, вы пишете скрипт на bash и хотите использовать в нем кон-струкцию if в следующем фрагменте: if [[...]] then echo "True" fi 47Вы можете вписать вместо “...” (внутри [[]]) и не забудьте

про пробелы 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. 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).(рис. 3.66).

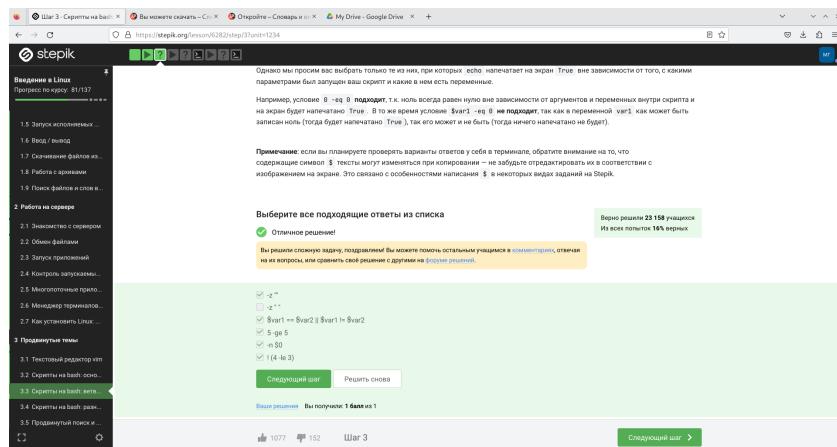


Рис. 3.66: Ответ на вопрос

12. 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.(рис. 3.67).

The screenshot shows a Stepik lesson titled "Шаг 5 - Скрипты на bash". In the top right, there's a progress bar indicating 100% completion. The main area contains a code editor with the following script:

```

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

```

Below the code, a note says: "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?"

A dropdown menu asks "Выберите один вариант из списка" with the option "Отмена" selected. Below it, three radio button options are shown:

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

Buttons at the bottom include "Следующий шаг" and "Решить снова". A progress bar at the bottom shows 1077 / 153 and "Шаг 5".

Рис. 3.67: Ответ на вопрос

13. 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.(рис. 3.68).

The screenshot shows a Stepik lesson titled "Шаг 6 - Скрипты на bash". In the top right, there's a progress bar indicating 85/137. The main area contains a code editor with the following script:

```

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 наше рекомендации по написанию скриптов.

Напишите программу. Тестируется через stdin -- stdout

```

A note above the code says: "Пример №2: если ваш скрипт называется ./script.sh , то при запуске его как ./script.sh 5 на экране должно появиться A lot of students".

A dropdown menu asks "Выберите один вариант из списка" with the option "Отмена" selected. Below it, three radio button options are shown:

- Тестовый ввод
- Аудитория
- Учебный ввод

Buttons at the bottom include "Следующий шаг" and "Решить снова". A progress bar at the bottom shows 85 / 137 and "Шаг 6".

Рис. 3.68: Ответ на вопрос

14. Если запустить этот скрипт, то сколько раз на экран будет выведено слово "start", а сколько раз слово "finish"? (рис. 3.69).

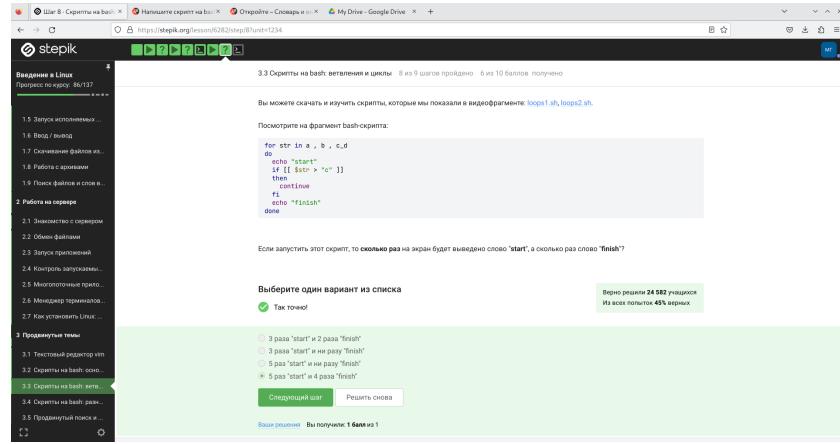


Рис. 3.69: Ответ на вопрос

15. Напишите скрипт на bash, который будет определять в какую возрастную группу попадают пользователи. При запуске скрипта должен вывести сообщение “enter your name:” и ждать от пользователя ввода имени (используйте read, чтобы прочитать его). Когда имя введено, скрипт должен написать “enter your age:” и ждать ввода возраста (опять нужен read). Когда возраст введен, скрипт пишет 51 на экран “, your group is”, где определяется на основе возраста по следующим правилам:(рис. 3.70).

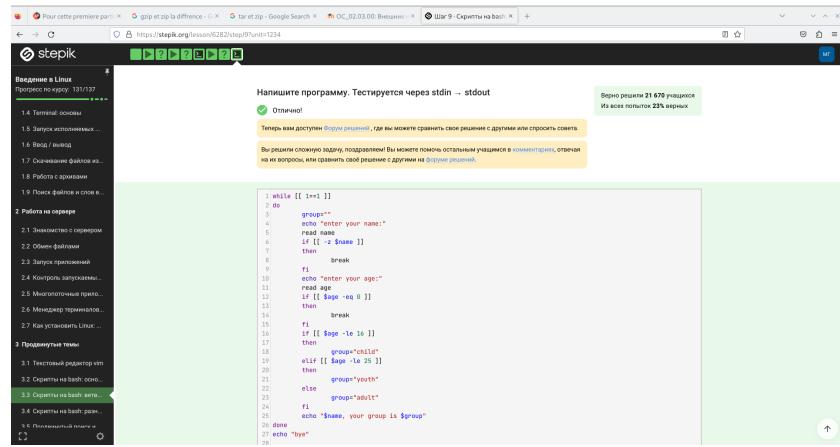


Рис. 3.70: Ответ на вопрос

16. Какие(ая) из предложенных ниже инструкций увеличивают значение перемен-

ной а 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!(рис. 3.71).

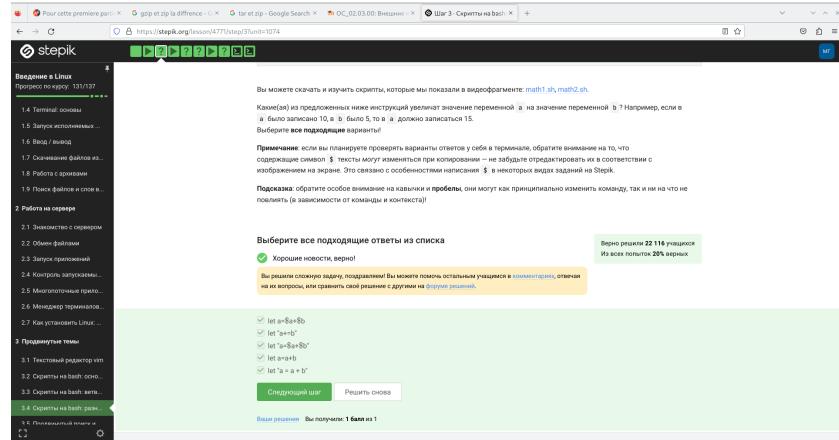


Рис. 3.71: Ответ на вопрос

17. Вы можете скачать и изучить скрипт, который мы показали в видеофрагменте: programs.sh.(рис. 3.72).

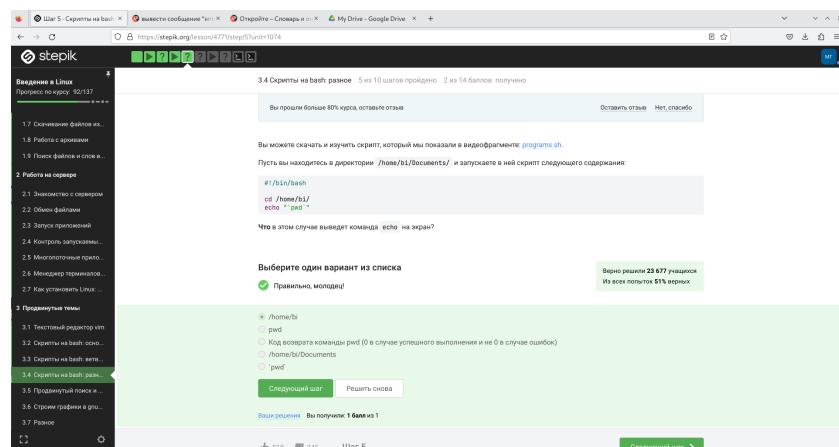


Рис. 3.72: Ответ на вопрос

18. 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 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 (”).(рис. 3.73).

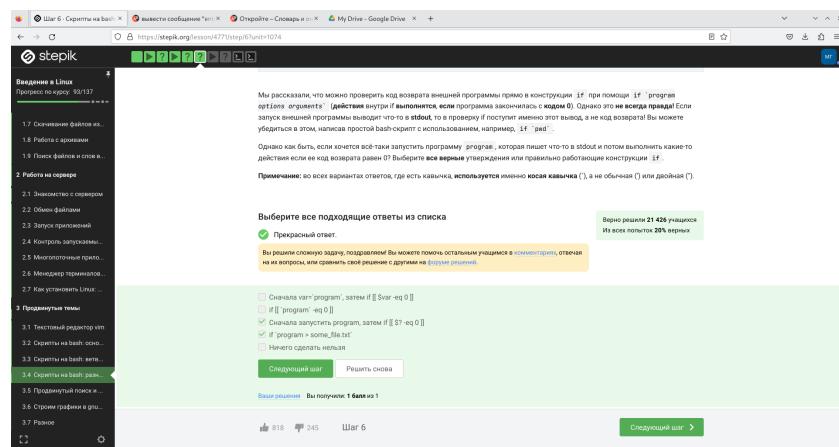


Рис. 3.73: Ответ на вопрос

19. 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.(рис. 3.74).

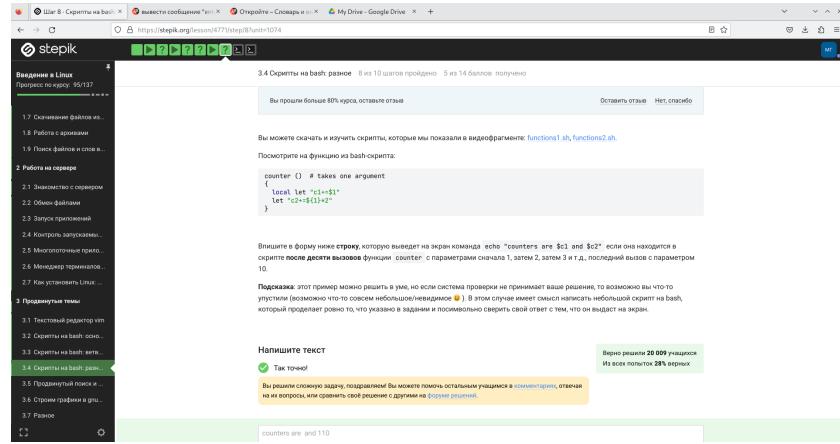


Рис. 3.74: Ответ на вопрос

20. 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”, 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.(рис. 3.75).

```

1 god O # M N
2 {
3     if [[ $1 == $2 -z $N ]]; then
4         echo "bye"
5         exit
6     fi
7     if [[ $1 == $N -m $N ]]; then
8         echo "$1"
9     else
10        if [[ $1 == $N -t $N ]]; then
11            let "M=$1"
12            god $M $N
13        else
14            let "N=$1"
15            god $M $N
16        fi
17    fi
18 }
19
20 while [[ true ]]; do
21     read M N
22     god $M $N
23 done
24
25
26
27
28
29

```

Рис. 3.75: Ответ на вопрос

21. 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 завершит Три аргумента через пробел – первый operand (целое число), операция (одна из “+”, “*”, “/”, “%”, “**”) и второй operand (целое число). В этом случае нужно произвести любая другая команда из одного аргумента или из трех аргументов, но с операцией н(рис. 3.76).)

```

1 #!/bin/bash
2 #!/bin/bash
3 while [[ ! True ]]
4 do
5     read Mirinchi; anal ikkinchi
6     if [[ $Mirinchi == "exit" ]]
7     then
8         echo "bye"
9         break
10    elif [[ "$Mirinchi" =~ ^[0-9]*$ && "$Ikkinchi" =~ ^[0-9]*$ ]]
11    then
12        echo "error"
13        break
14    else
15        case $anal in
16            "+") let "result = $Mirinchi + $Ikkinchi";;
17            "-") let "result = $Mirinchi - $Ikkinchi";;
18            "/") let "result = $Mirinchi / $Ikkinchi";;
19            "%") let "result = $Mirinchi % $Ikkinchi";;
20            "**") let "result = $Mirinchi ** $Ikkinchi";;
21            "+*") let "result = $Mirinchi * $Ikkinchi";;
22            "-*") let "result = $Mirinchi * $Ikkinchi";;
23            "error") break;;
24        esac
25        echo "$result"
26    fi
27 done
28
29
30

```

Рис. 3.76: Ответ на вопрос

22. Продвинутый поиск и редактирование 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"? (рис. 3.77).

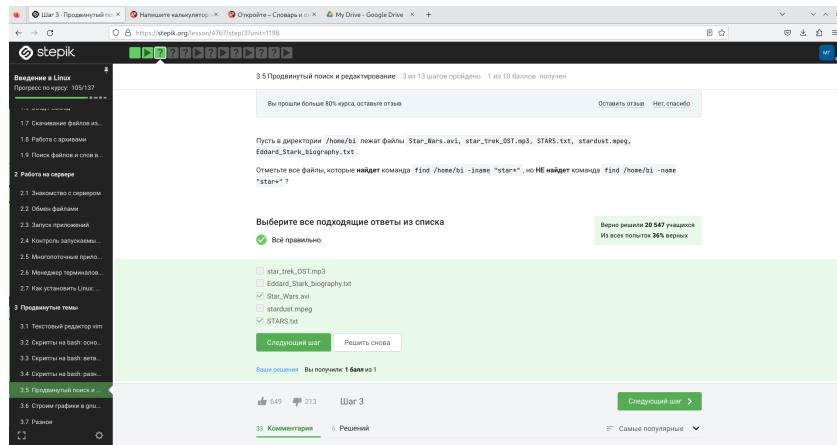


Рис. 3.77: Ответ на вопрос

23. Продвинутый поиск и редактирование 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"? (рис. 3.78).

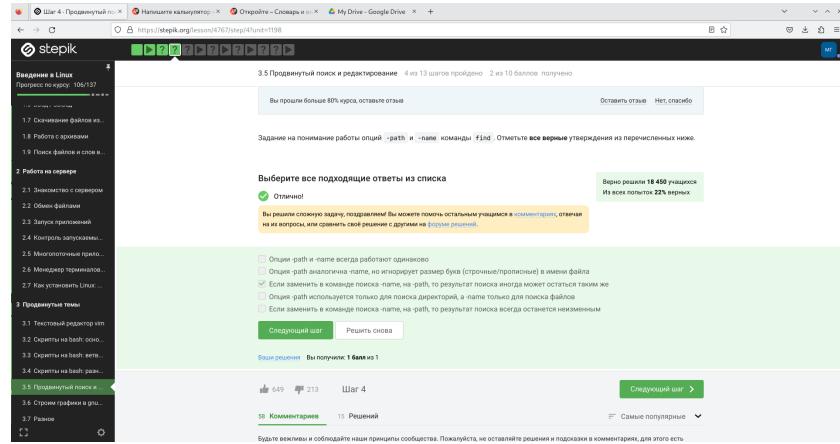


Рис. 3.78: Ответ на вопрос

24. Какие(ой) из трех файлов (file1, file2, file3) будут найдены по команде `find /home/bi -mindepth 2 -maxdepth 3 -name "file*"`? (рис. 3.79).

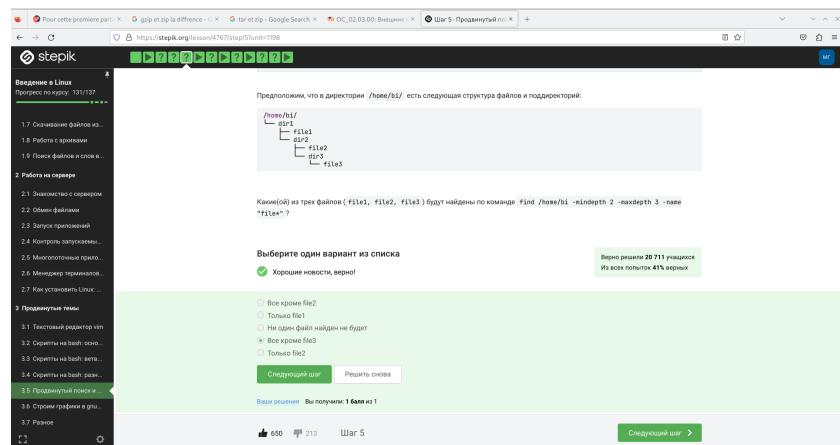


Рис. 3.79: Ответ на вопрос

25. 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:(рис. 3.80).

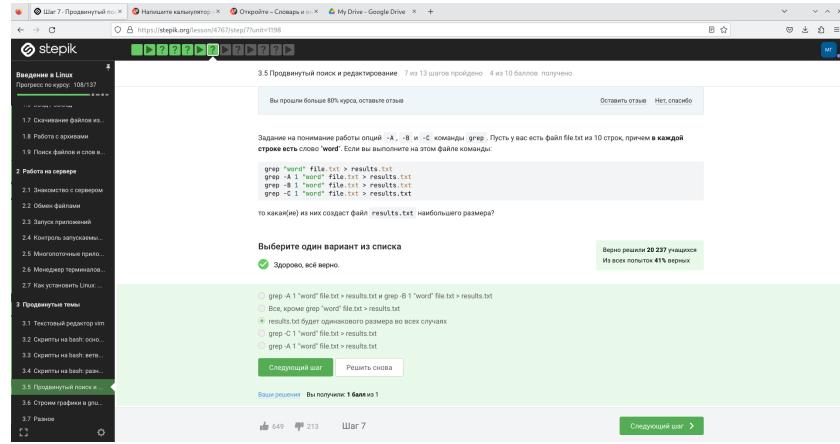


Рис. 3.80: Ответ на вопрос

26. 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.: (рис. 3.81).

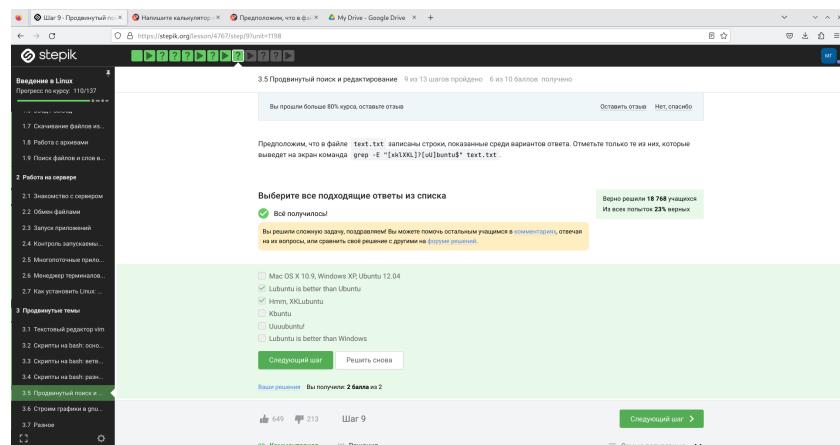


Рис. 3.81: Ответ на вопрос

27. Что произойдет, если в команде sed -n “/[a-z]*p” text.txt не указывать опцию -n(рис. 3.82).

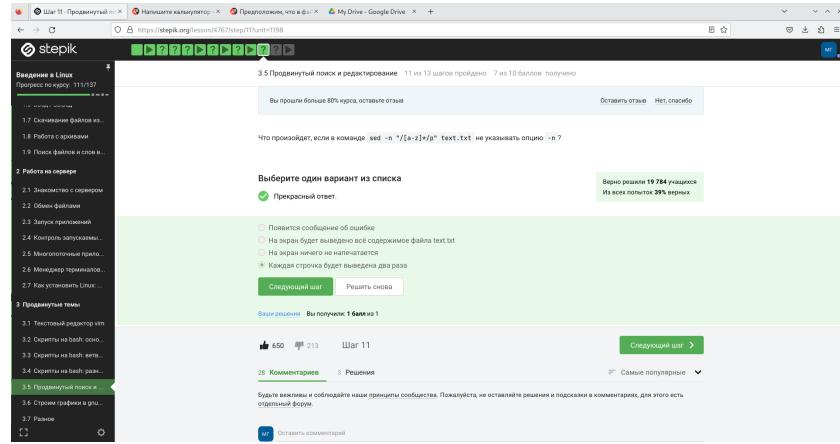


Рис. 3.82: Ответ на вопрос

28. Запишите в форму ниже инструкцию sed, которая заменит все “аббревиатуры” в файле input.txt на слово “abbreviation” и запишет результат в файл edited.txt (на экран при этом ничего выводить не нужно). Обратите внимание, что в ин- струкции должны быть указаны и сам sed, и оба файла!(рис. ??).

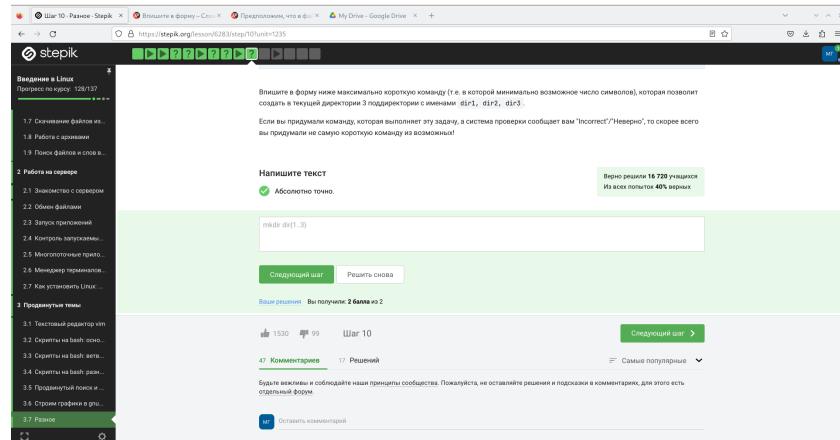


Рис. 3.83: Ответ на вопрос

29. Какую опцию нужно указать при запуске gnuplot, чтобы при его закрытии не были автоматически закрыты и все нарисованные в нём графики?(рис. ??).

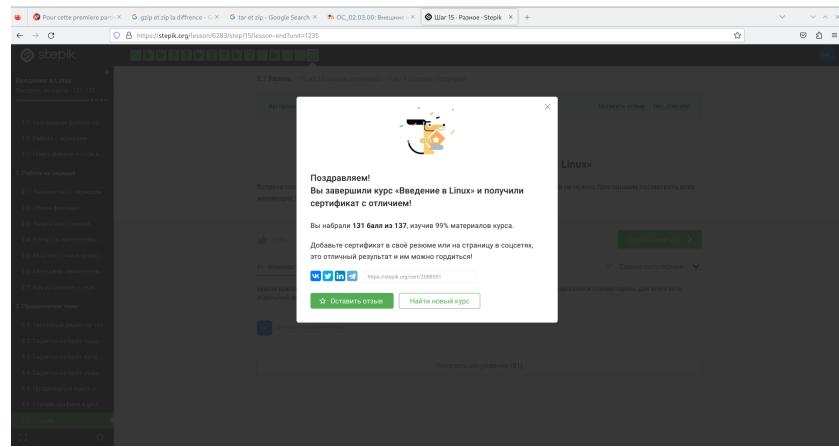


Рис. 3.84: Ответ на вопрос

30. Какое в этом случае будет название у построенного ряда данных и сколько будет нарисовано точек на графике?(рис. 3.85).

Рис. 3.85: Ответ на вопрос

31. 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 64черточки)

в этих самых координатах, а подписи этих делений были оформлены в виде “point , value”. Например, для $x_1=0$, $x_2=10$, $x_3=20$, 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, $x_1=100$, $x_2=150$, $x_3=250$, 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. (рис. 3.86).

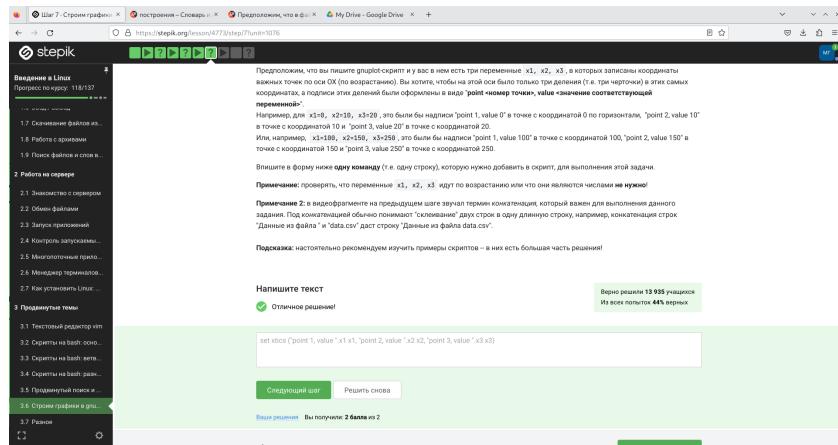


Рис. 3.86: Ответ на вопрос

32. Если вы не скачали на предыдущем шаге файлы `animated.gnu` и `move.rot`, то скачайте их теперь, т.к. они понадобятся для выполнения задания. Указанные файлы использовались в последнем видеофрагменте для создания врашающегося графика. Измените инструкции в файле `move.rot` (т.е. добавлять и удалять инструкции нельзя!) таким образом, чтобы: График отразился зеркально относительно горизонтальной поверхности. То есть там, 200), где была точка (-10, -10, 200) станет (-10, -10, -200) и т.д. При этом точ Изображение стало вращаться в обратную сторону. То есть если раньше вращалось “вл Вращение стало в два раза быстрее. То есть станет в два раза больше перерисовок г Измененный файл загрузите в форму ниже.(рис. 3.87).

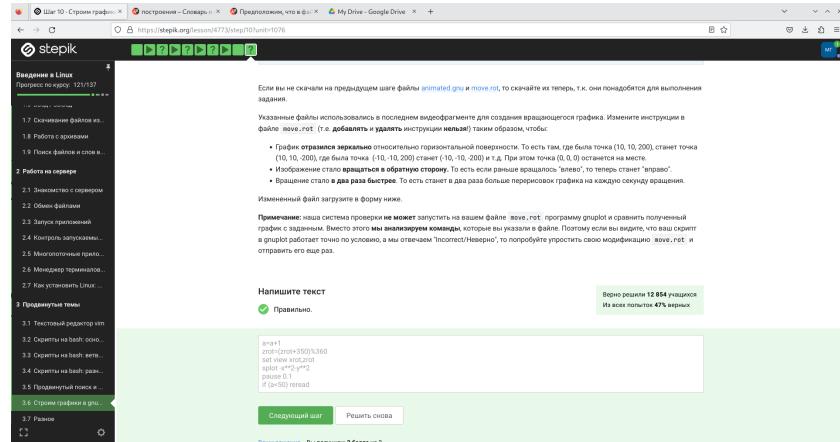


Рис. 3.87: Ответ на вопрос

33. Какая команда(ы) установят файлу file.txt права доступа rwxrw-r-, если изначально у него были права r-r-r-. Укажите все верные варианты ответа!
Примечание: запись вида команда1; команда2; команда3 означает, что в терминале последовательно выполнились все три команды (сначала команда1, затем команда2 и, наконец, команда3).(рис. 3.88).

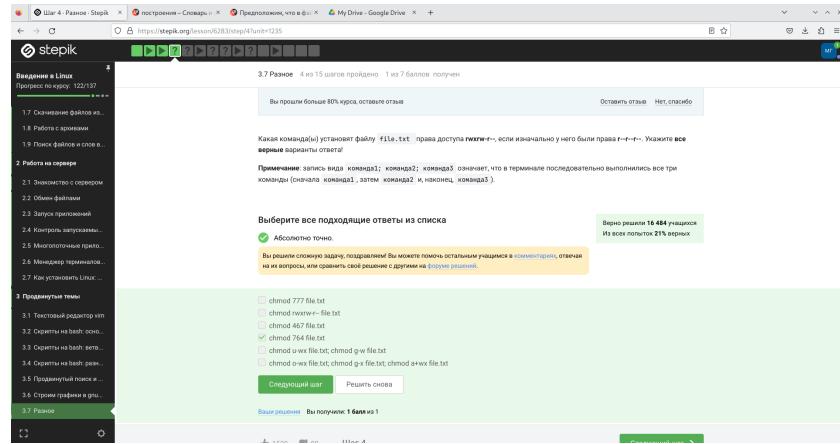


Рис. 3.88: Ответ на вопрос

34. Предположим вы использовали команду sudo для создания директории dir. По умолчанию для dir были выставлены права доступа rwxr-xr-x (владелец root, группа root). Таким образом никто кроме пользователя root не может

ничего записывать в эту директорию, например, не может создавать файлы в ней. После выполнения какой команды user из группы group всё-таки сможет создать файл внутри dir? Укажите все верные варианты ответов!(рис. 3.89).

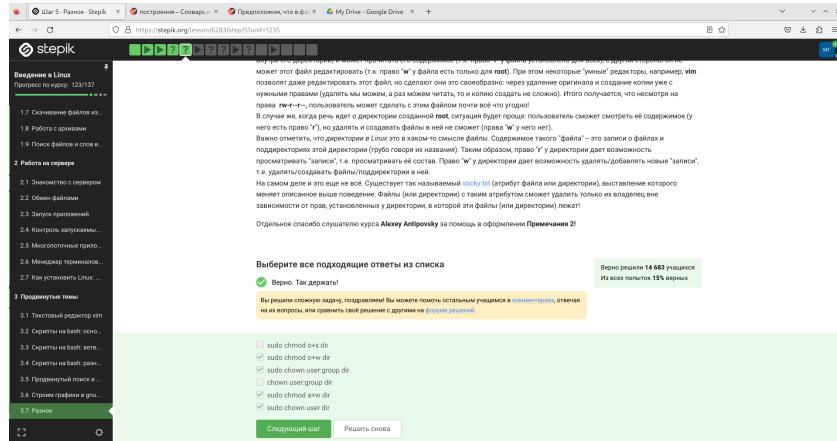


Рис. 3.89: Ответ на вопрос

35. Отметьте какие характеристики файла можно посчитать с использованием команды wc. (рис. 3.90).

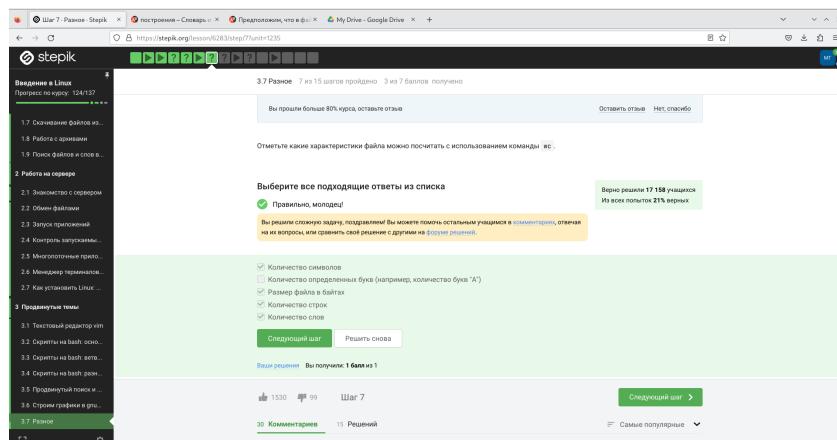


Рис. 3.90: Ответ на вопрос

36. 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 2048 байт надо выводить 2.0К) и больше на экран выводить ничего не нужно). В команде указывайте только необходимые для выполнения задания опции и аргументы, лишних опций указывать не нужно!(рис. 3.91).

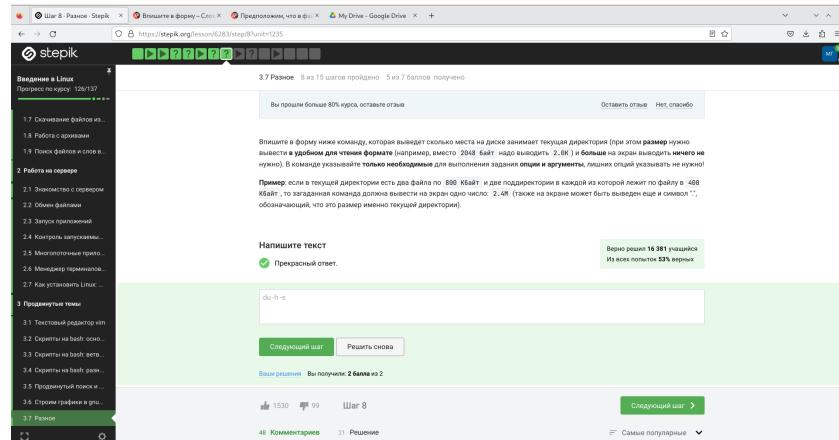


Рис. 3.91: Ответ на вопрос

37. 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!(рис. 3.92).

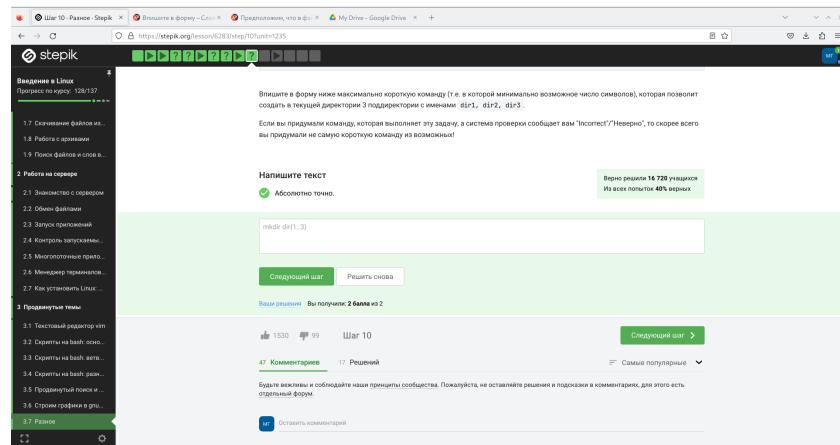


Рис. 3.92: Ответ на вопрос

38.

(рис. ??).

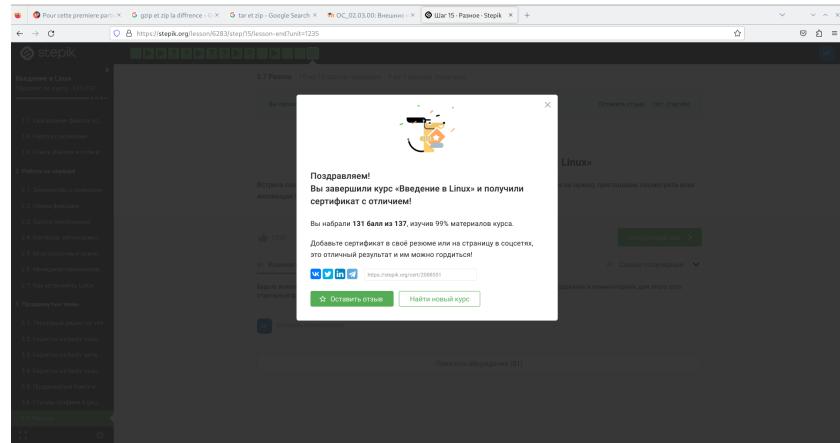


Рис. 3.93: Итоги курса

3.4 Сертификат

(рис. 3.94).

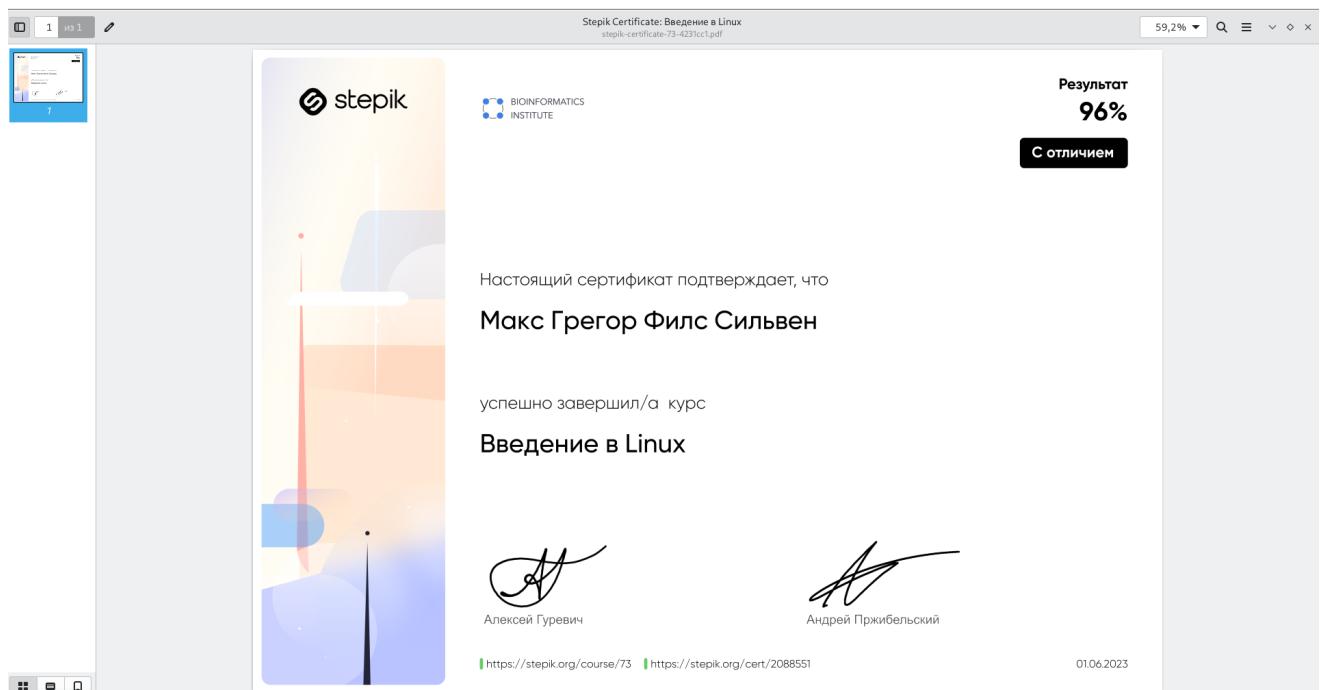


Рис. 3.94: Сертификат

4 Выводы

- В этой первой части мы изучили основы linux

Список литературы