Отчет о прохождении первого модуля внешнего курса

Введение в Linux. Модуль 2

Лазарев Даниил Михайлович

Содержание

| 1 | Цель работы | 5 |
|---|--------------------------------|----|
| 2 | Выполнение лабораторной работы | 6 |
| 3 | Выводы | 20 |

Список иллюстраций

| 2.1 | Задание 1. | • | • | • | • | • | • | • | • | • | | | • | | • | | | • | • | • | | • | | | • | 6 |
|------|-------------|---|---|---|---|---|---|---|---|---|---|--|---|---|---|---|---|---|---|---|---|---|---|---|---|----|
| 2.2 | Задание 2. | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 2.3 | Задание 3. | | | | | | | | | | | | | | | | | | | | | | | | | 7 |
| 2.4 | Задание 4. | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 2.5 | Задание 5. | | | | | | | | | | | | | | | | | | | | | | | | | 8 |
| 2.6 | Задание 6. | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 2.7 | Задание 7. | | | | | | | | | | | | | | | | | | | | | | | | | 9 |
| 2.8 | Задание 8 . | | | | | | | • | | • | | | | | | | | | | | | | | | | 10 |
| 2.9 | Задание 9. | | | | | | | • | | • | | | | | | | | | | | | | | | | 11 |
| 2.10 | Задание 10 | | | • | • | | | • | | • | | | • | | | • | | | | | • | | • | • | • | 12 |
| | Задание 11 | | | | | | | | | | | | | | | | | | | | | | | | | 12 |
| 2.12 | Задание 12 | • | | • | • | • | • | • | • | • | • | | | • | • | • | | | | • | | | | • | • | 13 |
| 2.13 | Задание 13 | • | | • | • | • | • | • | • | • | • | | | • | | • | | | | • | | | | • | • | 13 |
| | Задание 14 | | | | | | | | | | | | | | | | | | | | | | | | | 14 |
| 2.15 | Задание 15 | | • | • | • | • | • | • | • | • | • | | • | • | • | • | | • | • | • | • | | • | • | • | 14 |
| | Задание 16 | | | | | | | | | | | | | | | | | | | | | | | | | 15 |
| 2.17 | Задание 17 | | • | • | • | • | • | • | • | • | • | | • | • | • | • | | • | • | • | • | | • | • | • | 15 |
| 2.18 | Задание 18 | | | • | • | | | • | | • | | | • | | | • | | | | | • | | • | • | • | 16 |
| | Задание 19 | | | | | | | | | | | | | | | | | | | | | | | | | 16 |
| 2.20 | Задание 20 | | • | | | | | | | | | | | | | | • | | | | | | | | | 17 |
| 2.21 | Задание 21 | | | • | • | | | • | | • | | | • | | | • | | | | | • | | • | • | • | 17 |
| 2.22 | Задание 22 | | | | | | | • | | • | | | | | | | | | | | | | | | | 18 |
| 2.23 | Задание 23 | | | | • | | | | | | | | | | | | | | | | | | | | | 18 |
| 2 24 | Запание 24 | | | | | | | | | | | | | | | | | | | | | | | | | 19 |

Список таблиц

1 Цель работы

Пройти курс, который дополнит уже имеющиеся и возможно даст новые знания.

2 Выполнение лабораторной работы

Удаленный сервер уникальная вещь, поэтому может быть использован для практически любых задач. (рис. 2.1).

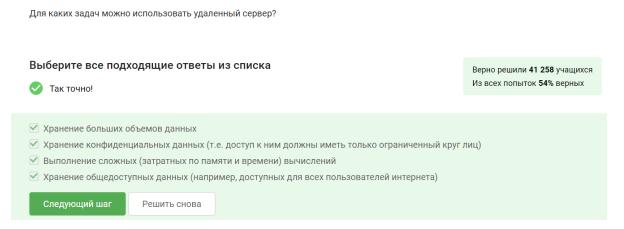


Рис. 2.1: Задание 1

Ключ имеет расширение pub от public, что изначально подразумевает, что ключ является публичным (рис. 2.2).

Предположим программа ssh-keygen создала вам два ключа: id_rsa и id_rsa.pub. Какой из этих ключей можно без опаски пересылать по интернету?

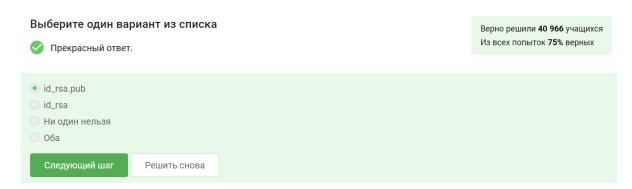


Рис. 2.2: Задание 2

Посмотрев видео перед заданием, мне удалось успешно его выполнить (рис. 2.3).

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

Выберите один вариант из списка

Верно решили 38 014 учащихся
Из всех попыток 57% верных

• scp -r stepic username@server:~/

scp stepic/* username@server:~/

ssh -cp stepic/* username@server:~/

ssh -cp stepic username@server:~/

Pешить снова

Рис. 2.3: Задание 3

Из банального: либо нет интернета, либо программа уже установлена, и нет возможности установить ее повторно, тогда придется использовать sudo apt-get update. (рис. 2.4).

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?

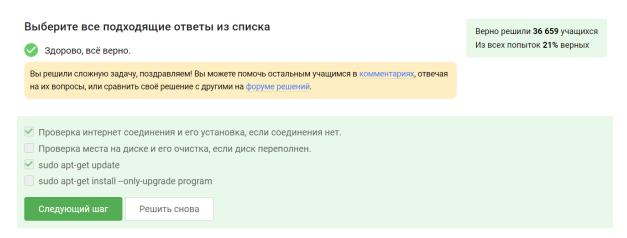


Рис. 2.4: Задание 4

Данную программу можно использовать практически для чего угодно, кроме установки программ на сервер. (рис. 2.5).

Выберите все подходящие ответы из списка

✓ Отличное решение!

✓ Для просмотра содержимого директорий на сервере

✓ Для просмотра содержимого директорий на своем компьютере

✓ Для установки программ на сервер

✓ Для копирования файлов с сервера на свой компьютере

✓ Для копирования файлов со своего компьютера на сервер

Рис. 2.5: Задание 5

Запускать на своем компьютере - путь слабаков, а чтобы ничего нельзя было сделать - такой ситуации не существует, я думаю. (рис. 2.6).

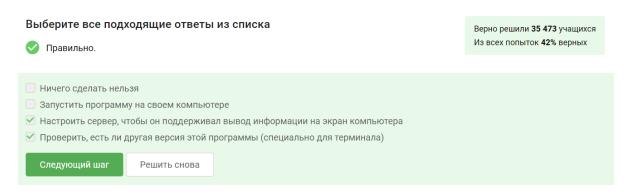


Рис. 2.6: Задание 6

Это самые стандартные способы узнать справку о программе, поэтому их и выбираем. (рис. 2.7).

Как обычно можно вызвать справочную информацию о программе program?

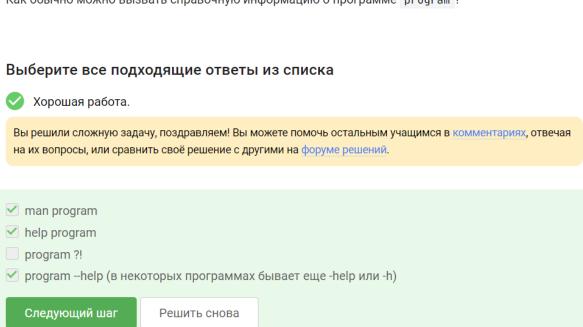


Рис. 2.7: Задание 7

Изучив справку для FastQC, я получила форматы, с которыми она может работать. (рис. 2.8).

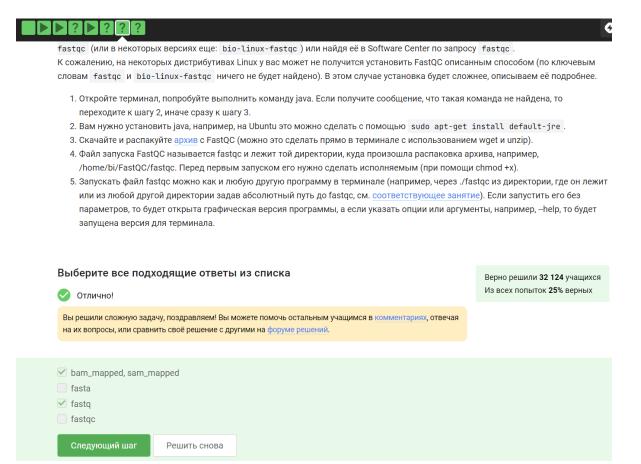


Рис. 2.8: Задание 8

Посмотрев справку, мне удалось сформировать команду, которая поможет в поставленной задаче. (рис. 2.9).

Clustal — это одна из самых широко используемых компьютерных программ для множественного выравнивания нуклеотидных и аминокислотных последовательностей (multiple sequence alignment). У нее есть графическая версия ClustalX и версия для запуска в терминале ClustalW. Вы можете потренироваться запускать его с использованием файла test.fasta.

Посмотрите справку по программе (имеется в виду 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)!

Примечание: справку по опциям можно получить при помощи man или, если он у вас не работает, то в разделе "Help for command line parameters" файла clustalw_help.txt, который идет в поставке программы.

Примечание 2: программа Clustal запускает необходимый алгоритм выравнивания по умолчанию (т.е. если ему не указать каких-либо других опций), однако мы просим вас найти и **указать** в команде запуска **опцию**, которая явно говорит Clustal запустить именно множественное выравнивание. После этого вы можете сравнить вывод Clustal при запуске с этой опцией и без нее – результат должен быть одинаков.

Подсказка: если у вас не установлена программа Clustal, то её можно установить командой sudo apt-get install clustalw (или clustalx) или найдя её в Software Center по запросу clustalw (clustalx). Обратите внимание, что на некоторых дистрибутивах доступна только вторая версия программы (например, clustalw2), в этом случае можете использовать и её — все необходимые в задании опции будут точно такими же.



Рис. 2.9: Задание 9

Для достоверности была воспроизведена такая же ситуация на моей виртуальной машине. (рис. 2.10).

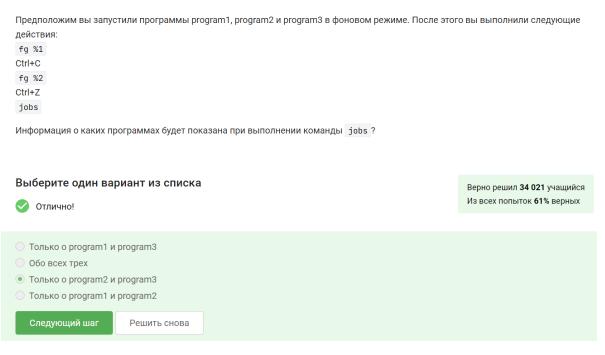


Рис. 2.10: Задание 10

Информация была получена из справки по команде kill. (рис. 2.11).

jobs , top и ps позволяют отслеживать работу запущенных в терминале программ. В каждой из этих трех утилит для каждой запущенной программы указывается число-идентификатор. Одинаковые ли эти идентификаторы в jobs , top и ps ?



Рис. 2.11: Задание 11

Выявлено практическим путём. (рис. 2.12).

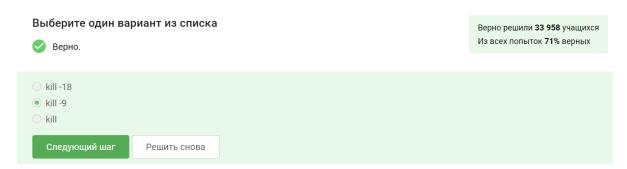


Рис. 2.12: Задание 12

Это было также выявлено практическим путём. (рис. 2.13).

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

Выберите один вариант из списка

Правильно.

Верно решили 33 808 учащихся Из всех попыток 47% верных

После этого действия процесс невозможно будет вернуть к работе

Процесс приступит к завершению, как только будет продолжен

Это никак не повлияет на процесс

Процесс будет завершен

Следующий шаг

Решить снова

Рис. 2.13: Задание 13

Такое приложение использует только память, чтобы оставаться в фоне, что будет показано в следующем задании. Поэтому процессор никаким образом не будет загружен этим приложением. (рис. 2.14).

Сколько вычислительных ресурсов центрального процессора (% CPU) использует остановленное (по Ctrl+Z) многопоточное 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% СРU, однако наш вопрос касается именно момента после остановки такого приложения. 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. Подсказка 2: подробнее почитать о значении всей информации, которую выводит top на экран, можно по ссылке http://ruslinux.net/MyLDP/consol/komanda-top-v-linux.html Выберите один вариант из списка Верно решили 32 397 учащихся Из всех попыток 59% верных Всё правильно. О Столько, сколько использовалось до остановки 🔘 В два раза меньше, чем использовалось до остановки 0% CPU ○ 100% CPU

Рис. 2.14: Задание 14

Подтверждение предыдущему пояснению. (рис. 2.15).

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.

Подсказка 2: подробнее почитать о значении всей информации, которую выводит top на экран, можно по ссылке http://rus-linux.net/MyLDP/consol/komanda-top-v-linux.html

Выберите один вариант из списка Отличное решение! Отличное решение!

Рис. 2.15: Задание 15

Отдельный поток многопоточного приложения остановить не получится. (рис. 2.16).

Как принудительно завершить один из потоков запущенного многопоточного приложения?

Подсказка: если вы не знаете как ответить на этот вопрос, то можете попробовать запустить многопоточное приложение на своем компьютере и посмотреть на результат. 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.

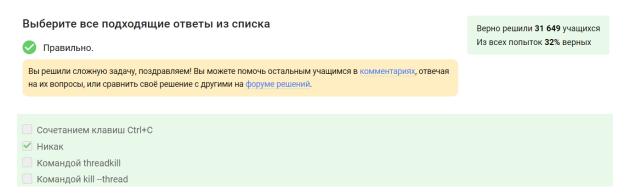


Рис. 2.16: Задание 16

Изучено практическим путём. (рис. 2.17).

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?

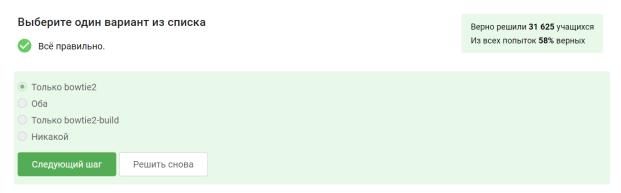


Рис. 2.17: Задание 17

Код программы был получен практическим путём. (рис. 2.18).

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 количеству ядер на вашем компьютере (команда пргос). 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!

Примечание: если у вас не очень сильный компьютер, то работа bowtie2 на предложенных данных может занять достаточно продолжительное время. Если вы не хотите ждать, то можете использовать альтернативные (сильно уменьшенные) версии референсного генома (reference) и ридов (reads). На этих данных у вас не получится увидеть разницу в скорости при запуске в один или в несколько потоков, но вы сможете выполнить все остальные пункты задания и получить за него полный балл.



Рис. 2.18: Задание 18

Ответ был получен также практическим путём. (рис. 2.19).

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:

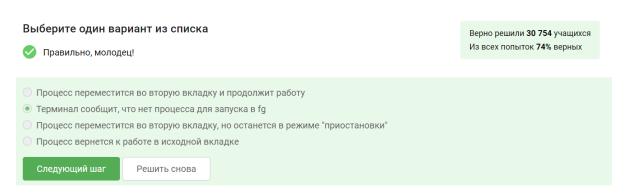


Рис. 2.19: Задание 19

Ответ всё еще был получен практическим путём. (рис. 2.20).

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



Рис. 2.20: Задание 20

Информация, найденная в интернете и подкрепленная практическим путём. (рис. 2.21).

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?

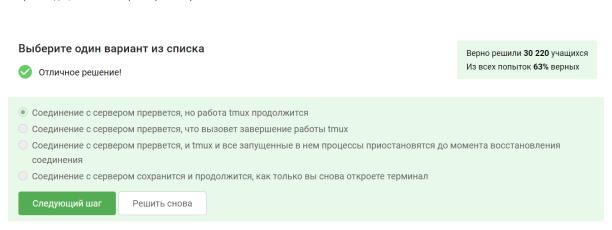


Рис. 2.21: Задание 21

Ответ получен практическим путём. (рис. 2.22).

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



Рис. 2.22: Задание 22

Изучив справку по команде, был получен ответ. (рис. 2.23).

| Задание на самостоятельное изучение tmux. | | | | | | | | |
|---|---|--|--|--|--|--|--|--|
| Изучите справку по tmux (например, man tmux) и выберите из предложенных ниже tmux-команд ту, которая отвечает за переименование текущей вкладки. | | | | | | | | |
| Выберите один вариант из списка | Верно решили 29 445 учащихся Из всех попыток 54 % верных | | | | | | | |
| Сtrl+B и і Сtrl+B и . (точка) € Ctrl+B и , (запятая) Сtrl+B и 0 Сtrl+B и t | | | | | | | | |
| Следующий шаг Решить снова | | | | | | | | |

Рис. 2.23: Задание 23

Как и предыдущее задание - ответ был получен после изучения справки по команде. (рис. 2.24).

Задание на самостоятельное изучение tmux.

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 утверждениях у себя на компьютере.

Выберите все подходящие ответы из списка ✓ Правильно. Вы решили сложную задачу, поздравляем! Вы можете помочь остальным учащимся в комментариях, отвечая на их вопросы, или сравнить своё решение с другими на форуме решений. □ Если набрать в одной из "частей" вкладки команду exit, то вся вкладка закроется ✓ Вкладку можно разделить и горизонтально, и вертикально, и даже по несколько раз — просто используем нужные команды"разделения" необходимое количество раз ✓ Можно закрыть одну из "частей" вкладки выполнив (Ctrl+B и x) ✓ Если разделенную горизонтально вкладку разделить еще и вертикально (т.е. нажать один раз Ctrl+B и %), то получится 3 "части" — две маленькие и одна большая ✓ Команды-"разделения" действуют только в текущей вкладке tmux, а не во всех вкладках одновременно □ Вкладку можно разделить только горизонтально или только вертикально, а на попытку ввести вторую команду-"разделения" она реагировать уже не будет

Рис. 2.24: Задание 24

3 Выводы

Мы прошли второй модуль внешнего курса "Введение в Linux" и освежили уже имеющиеся знания, а также получили некоторые новые.