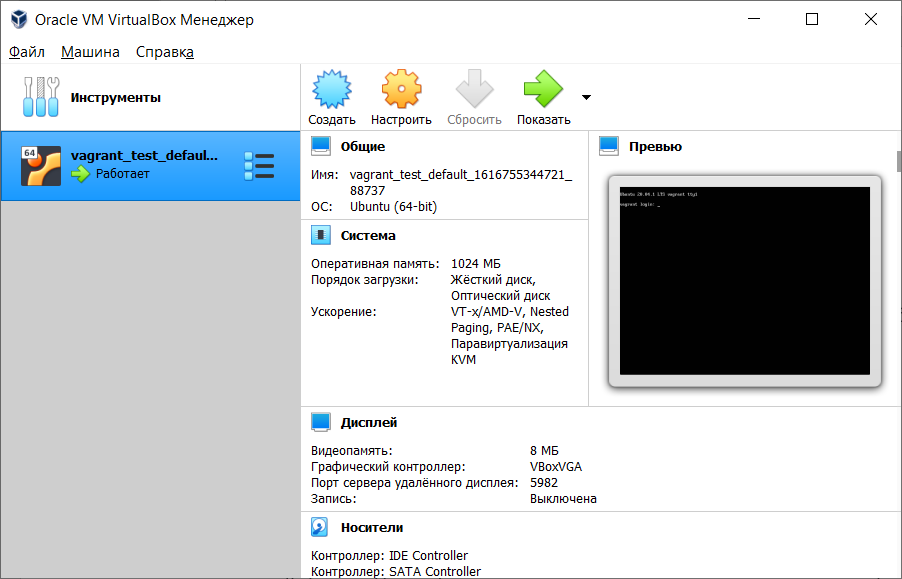
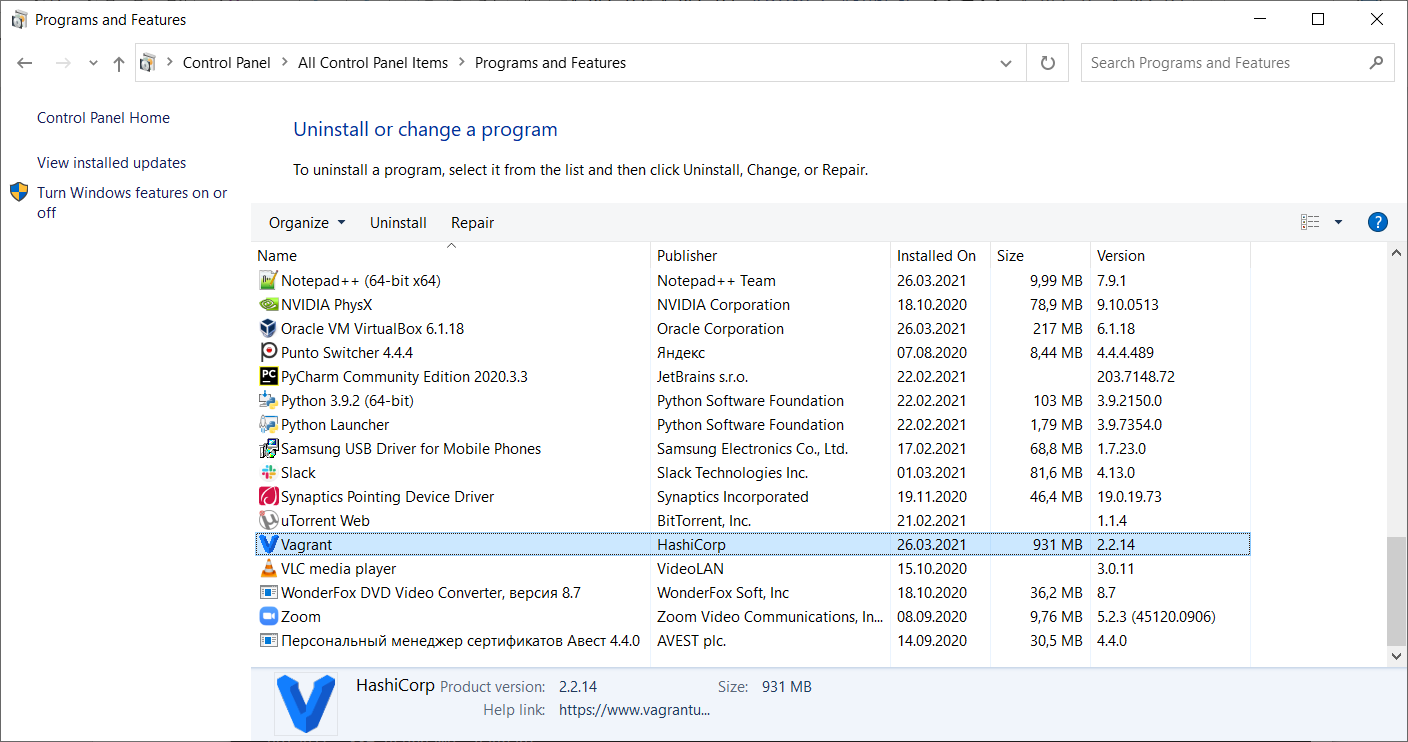
1. Установите средство виртуализации [Oracle VirtualBox](https://www.virtualbox.org/).



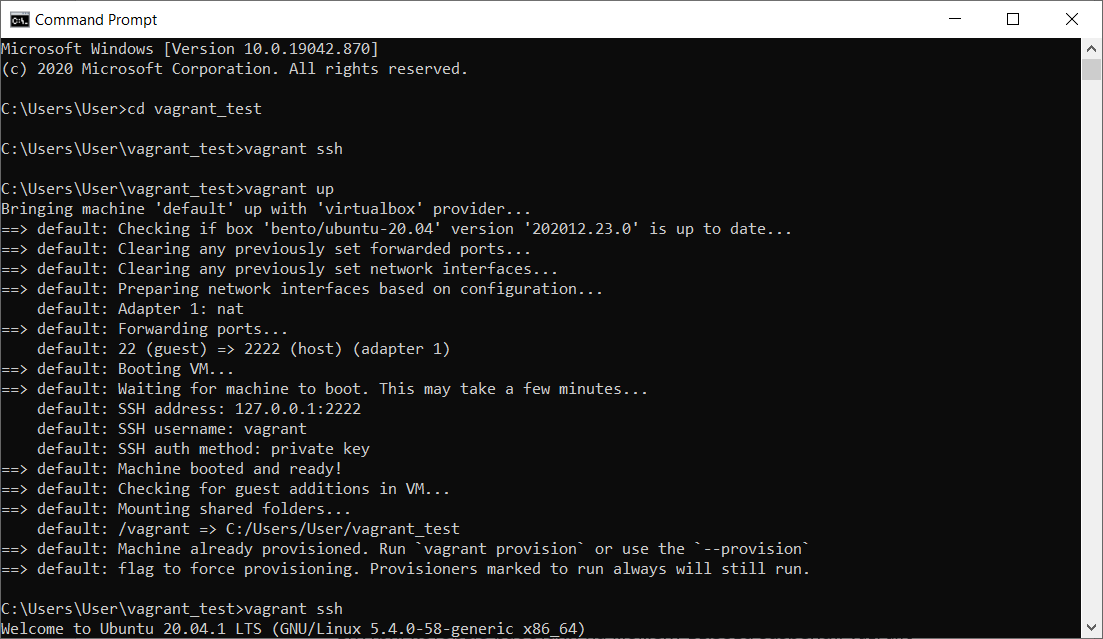
1. Установите средство автоматизации [Hashicorp Vagrant](https://www.vagrantup.com/).



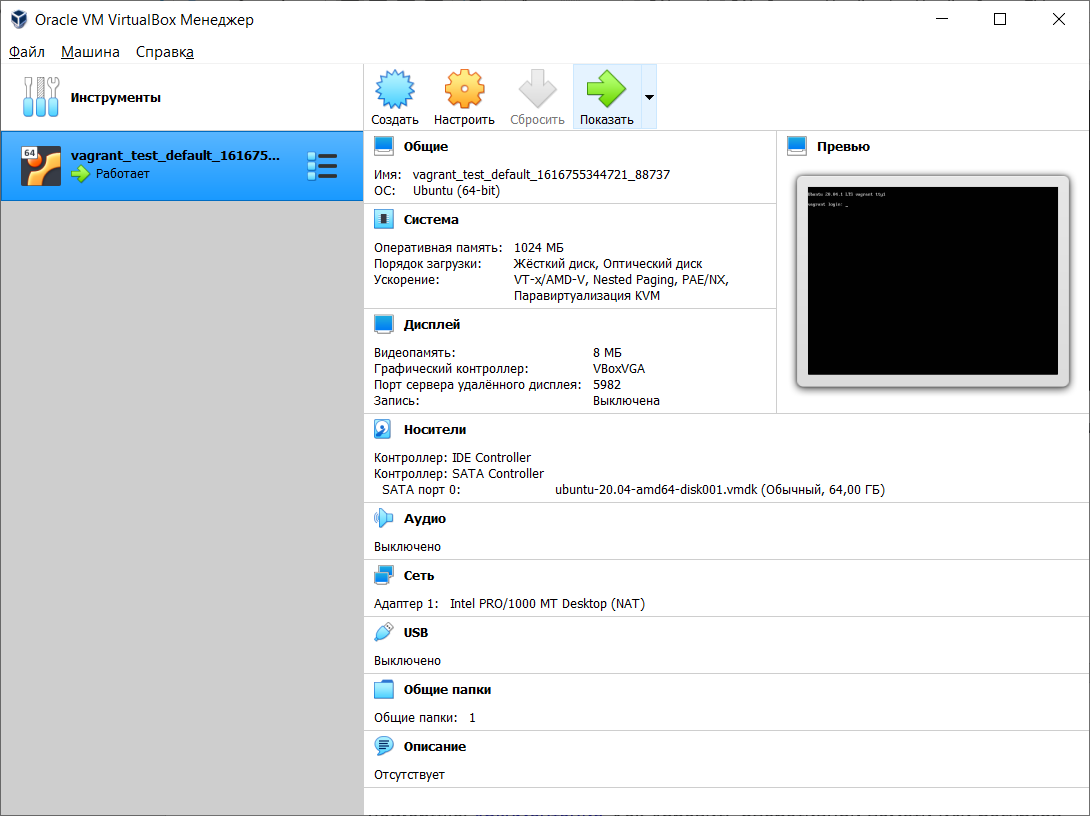
1. В вашем основном окружении подготовьте удобный для дальнейшей работы терминал.
2. С помощью базового файла конфигурации запустите Ubuntu 20.04 в VirtualBox посредством Vagrant:
   * Создайте директорию, в которой будут храниться конфигурационные файлы Vagrant. В ней выполните vagrant init. Замените содержимое Vagrantfile по умолчанию следующим:
   * Vagrant.configure("2") do |config|
   * config.vm.box = "bento/ubuntu-20.04"

end

* + Выполнение в этой директории vagrant up установит провайдер VirtualBox для Vagrant, скачает необходимый образ и запустит виртуальную машину.
  + vagrant suspend выключит виртуальную машину с сохранением ее состояния (т.е., при следующем vagrant up будут запущены все процессы внутри, которые работали на момент вызова suspend), vagrant halt выключит виртуальную машину штатным образом.



1. Ознакомьтесь с графическим интерфейсом VirtualBox, посмотрите как выглядит виртуальная машина, которую создал для вас Vagrant, какие аппаратные ресурсы ей выделены. Какие ресурсы выделены по-умолчанию?



1. Ознакомьтесь с возможностями конфигурации VirtualBox через Vagrantfile: [документация](https://www.vagrantup.com/docs/providers/virtualbox/configuration.html). Как добавить оперативной памяти или ресурсов процессора виртуальной машине?

Vagrant.configure("2") do |config|

config.vm.box = "centos/7"

config.vm.provider "virtualbox" do |vb|

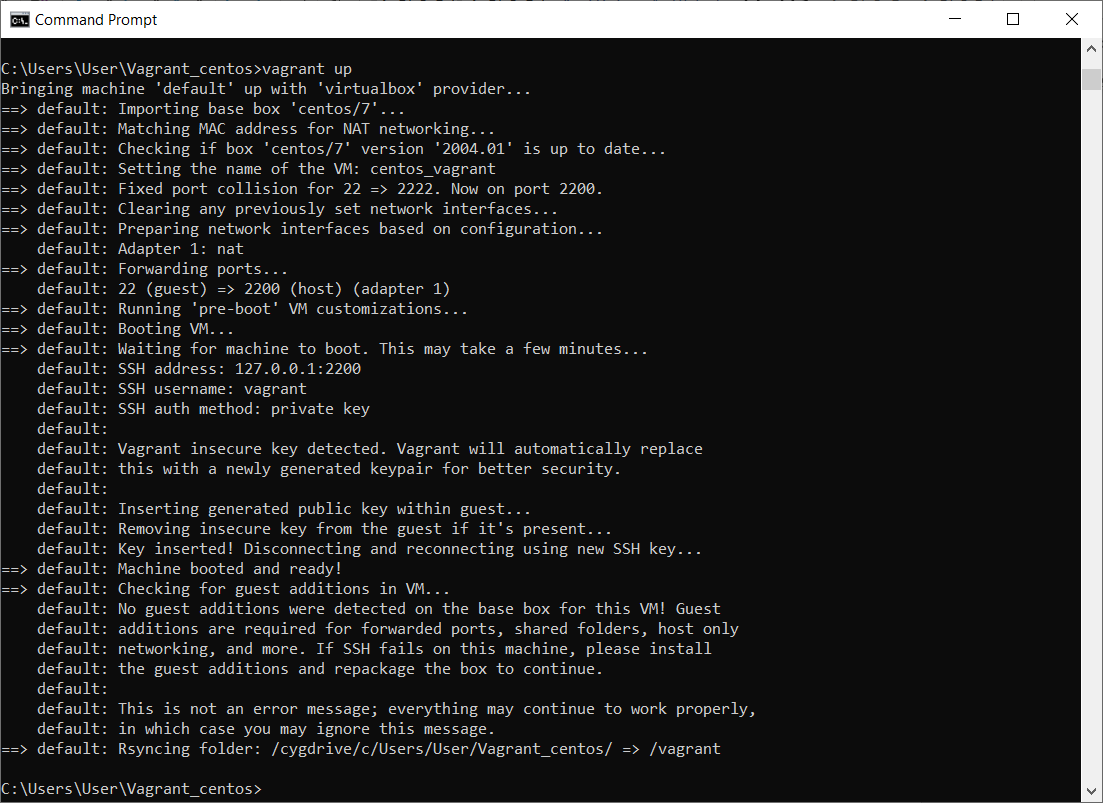
vb.name = "centos\_vagrant"

vb.memory = 1024

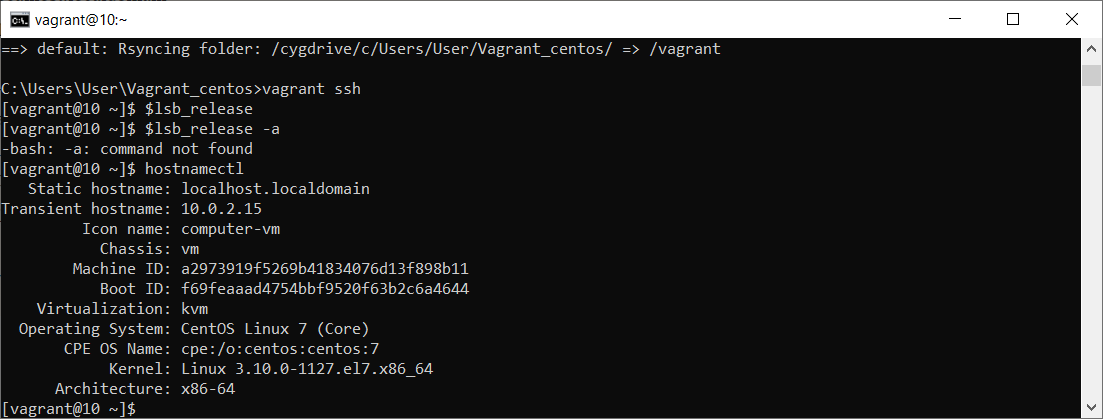
vb.cpus = 2

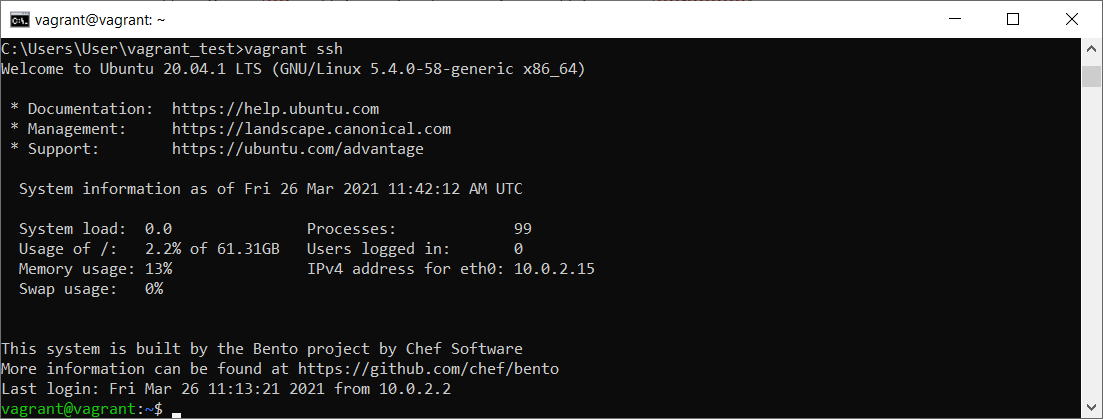
end

end

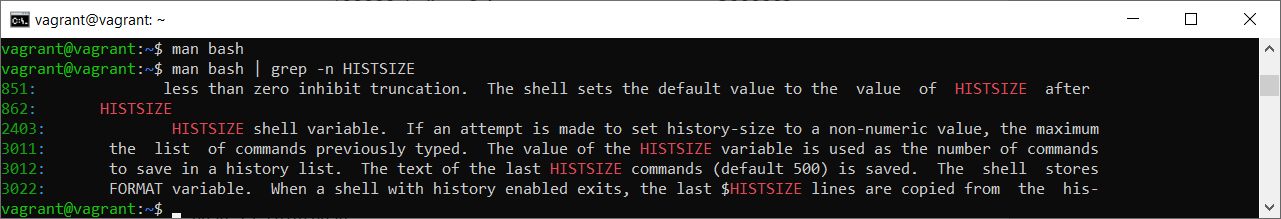


1. Команда vagrant ssh из директории, в которой содержится Vagrantfile, позволит вам оказаться внутри виртуальной машины без каких-либо дополнительных настроек. Попрактикуйтесь в выполнении обсуждаемых команд в терминале Ubuntu.

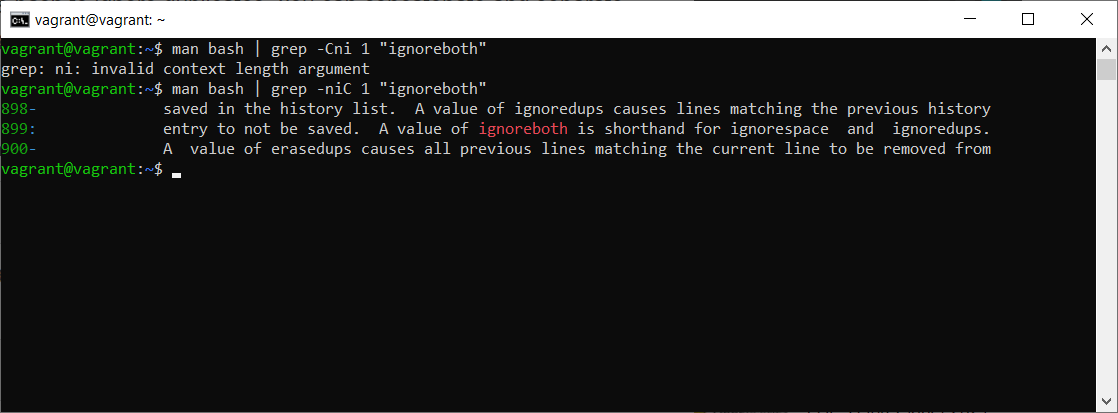




1. Ознакомиться с разделами man bash, почитать о настройках самого bash:
   * какой переменной можно задать длину журнала history, и на какой строчке manual это описывается?



* + что делает директива ignoreboth в bash?

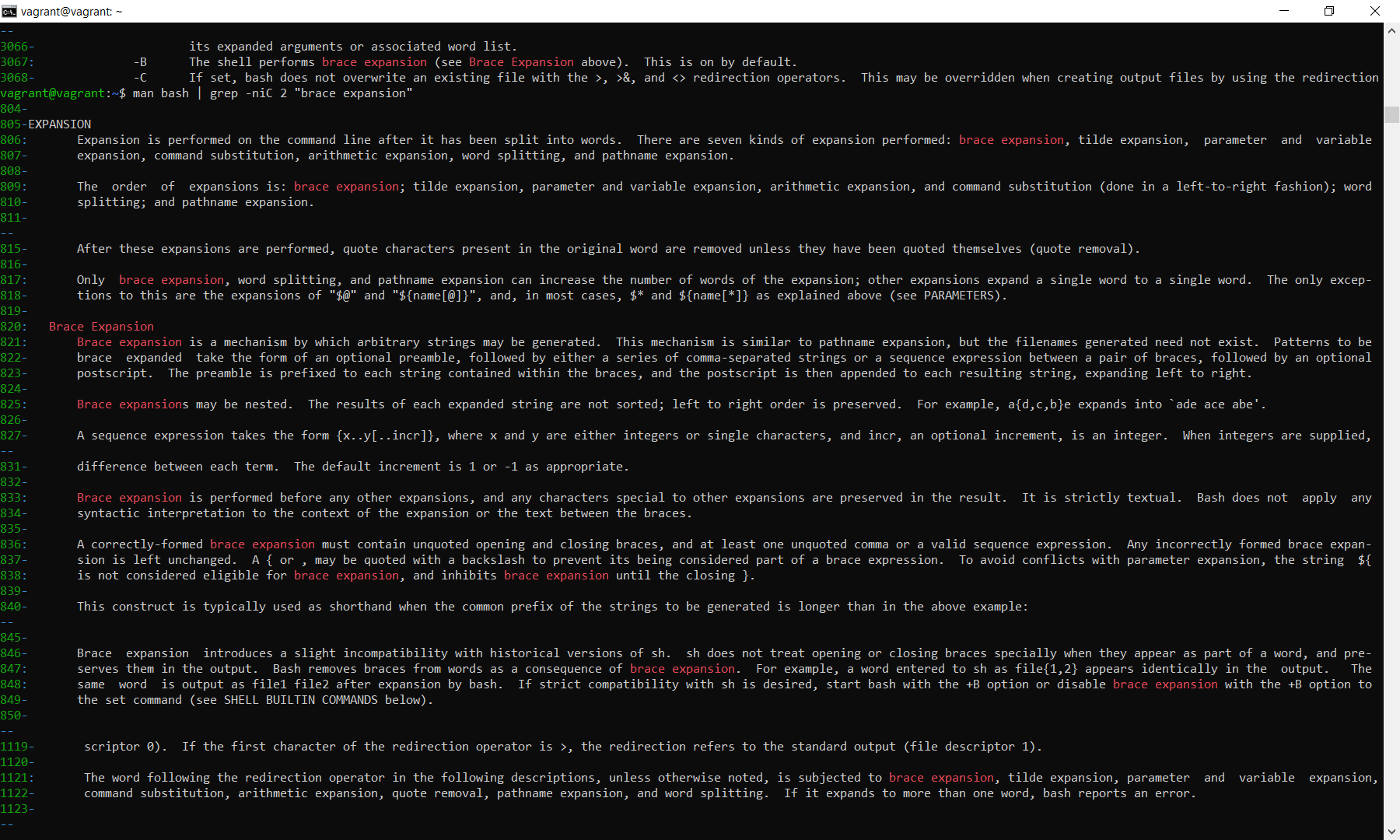


1. В каких сценариях использования применимы скобки {} и на какой строчке man bash это описано?

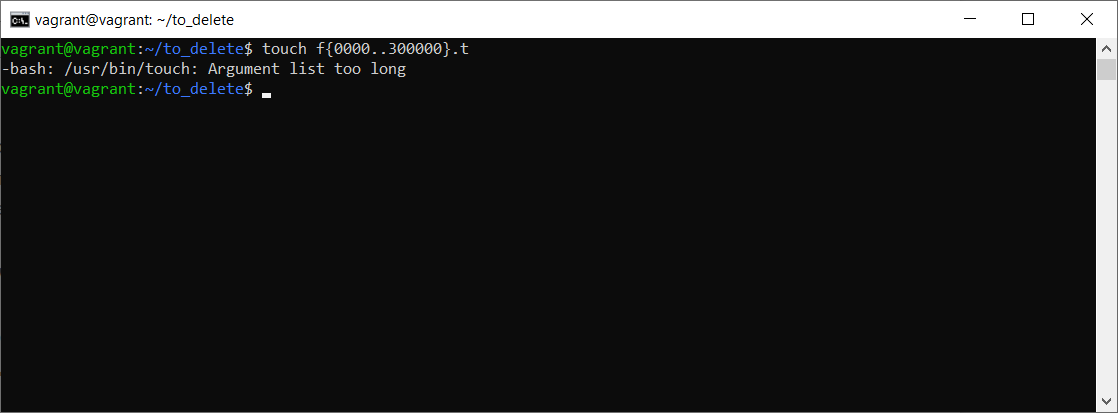
Для вывод всех вхождений использования фигурных скобок я бы использовал

man bash | grep -niC 1 "{"

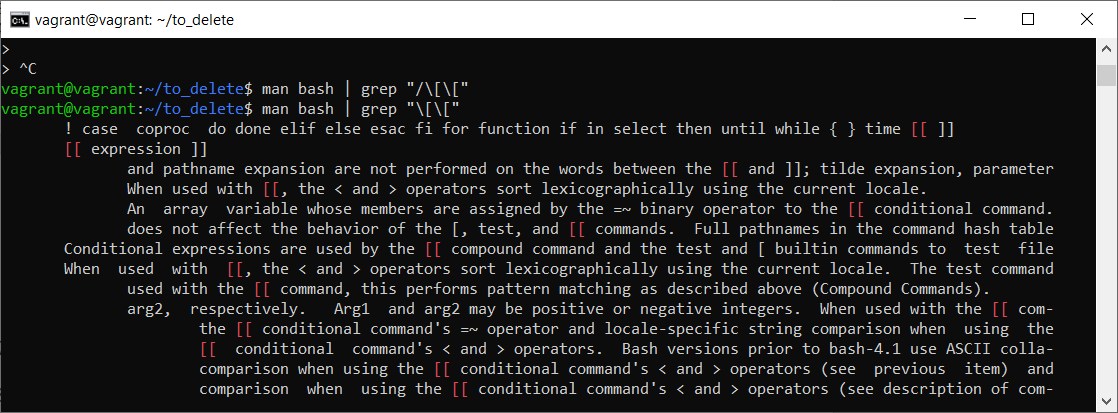
Однако вывод подобной команды обширный и слабочитаемый, однако на основании следующего задания могу предположить, что речь идет о brace expansion



1. Основываясь на предыдущем вопросе, как создать однократным вызовом touch 100000 файлов? А получилось ли создать 300000?



1. В man bash поищите по /\[\[.



Что делает конструкция [[ -d /tmp ]]

**-d *file***

True if *file* exists and is a directory.

Возвращает true если в папке есть файлы. Регулярное выражение используется для очистки папки темп.

1. Основываясь на знаниях о просмотре текущих (например, PATH) и установке новых переменных; командах, которые мы рассматривали, добейтесь в выводе type -a bash в виртуальной машине наличия первым пунктом в списке:

bash is /tmp/new\_path\_directory/bash

bash is /usr/local/bin/bash

bash is /bin/bash

(прочие строки могут отличаться содержимым и порядком)

Это было сложно, я практически сдался.

[vagrant@10 ~]$ export PATH="/tmp/new\_path\_directory/:$PATH"

[vagrant@10 ~]$ type -a bash

bash is /tmp/new\_path\_directory/bash

bash is /usr/bin/bash

bash is /tmp/new\_path\_directory/bash – это симилинк на баш добавленный в PATH

1. Чем отличается планирование команд с помощью batch и at?

At используется для назначения разовых задач по таймеру. А batch – разовых задач, когда загрузка становится меньше 0.8

1. Завершите работу виртуальной машины чтобы не расходовать ресурсы компьютера и/или батарею ноутбука.

