

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1

**УСТАНОВКА И КОНФИГУРАЦИЯ ОПЕРАЦИОННОЙ
СИСТЕМЫ НА ВИРТУАЛЬНУЮ МАШИНУ**

дисциплина: Архитектура компьютера

Студент: Обрезкова Анастасия Владимировна

Группа: НММбд-02-22

МОСКВА

2022

Оглавление.

1. Цель работы.	3
2. Задание.	3
3. Теоретическое введение.	3
3.1. Введение в командную строку GNU Linux.	4
4. Выполнение лабораторной работы.	7
5. Самостоятельная работа.	16
6. Вывод.	20
7. Список литературы.	20
8. Список иллюстраций.	21

1. Цель работы.

Целью данной работы является приобретение практических навыков установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

2. Задание.

Установить и настроить виртуальную машину с операционной системой Linux.

3. Теоретическое введение.

Введение в GNU Linux Операционная система (ОС) — это комплекс взаимосвязанных программ, предназначенных для управления ресурсами компьютера и организации взаимодействия с пользователем. Сегодня наиболее известными операционными системами являются ОС семейства Microsoft Windows и UNIX-подобные системы.

GNU Linux — семейство переносимых, многозадачных и многопользовательских операционных систем, на базе ядра Linux, включающих тот или иной набор утилит и программ проекта GNU, и, возможно, другие компоненты. Как и ядро Linux, системы на его основе, как правило, создаются и распространяются в соответствии с моделью разработки свободного и открытого программного обеспечения (Open-Source Software). Linux-системы распространяются в основном бесплатно в виде различных дистрибутивов.

Дистрибутив GNU Linux — общее определение ОС, использующих ядро Linux и набор библиотек и утилит, выпускаемых в рамках проекта GNU,

а также графическую оконную подсистему X Window System. Дистрибутив готов для конечной установки на пользовательское оборудование. Кроме ядра и, собственно, операционной системы дистрибутивы обычно содержат широкий набор приложений, таких как редакторы документов и таблиц, мультимедийные проигрыватели, системы для работы с базами данных и т.д. Существуют дистрибутивы, разрабатываемые как при коммерческой поддержке (Red Hat / Fedora, SLED / OpenSUSE, Ubuntu), так и исключительно усилиями добровольцев (Debian, Slackware, Gentoo, ArchLinux).

3.1. Введение в командную строку GNU Linux.

Работу ОС GNU Linux можно представить в виде функционирования множества взаимосвязанных процессов. При загрузке системы сначала запускается ядро, которое, в свою очередь, запускает оболочку ОС (от англ. shell «оболочка»). Взаимодействие пользователя с системой Linux (работа с данными и управление работающими в системе процессами) происходит в интерактивном режиме посредством командного языка. Оболочка операционной системы (или командная оболочка, интерпретатор команд) — интерпретирует (т.е. переводит на машинный язык) вводимые пользователем команды, запускает соответствующие программы (процессы), формирует и выводит ответные сообщения. Кроме того, на языке командной оболочки можно писать небольшие программы для выполнения ряда последовательных операций с файлами и содержащимися в них данными — сценарии (скрипты).

Из командных оболочек GNU Linux наиболее популярны bash, csh, ksh, zsh. Команда `echo $SHELL` позволяет проверить, какая оболочка используется. В качестве предустановленной командной оболочки GNU Linux используется одна из наиболее распространённых разновидностей командной оболочки — bash (Bourne again shell).

В GNU Linux доступ пользователя к командной оболочке обеспечивается через терминал (или консоль). Запуск терминала можно осуществить через главное меню «Приложения» «Стандартные» «Терминал (или Консоль)» или нажав Ctrl + Alt + t . Интерфейс командной оболочки очень прост. Обычно он состоит из приглашения командной строки (строки, оканчивающейся символом \$), по которому пользователь вводит команды:

```
iivanova@dk4n31:~$
```

Это приглашение командной оболочки, которое несёт в себе информацию об имени пользователя iivanova, имени компьютера dk4n31 и текущем каталоге, в котором находится пользователь, в данном случае это домашний каталог пользователя, обозначенный как ~).

Команды могут быть использованы с ключами (или опциями) — указаниями, модифицирующими поведение команды. Ключи обычно начинаются с символа (-) или (--) и часто состоят из одной буквы. Кроме ключей после команды могут быть использованы аргументы (параметры) — названия объектов, для которых нужно выполнить команду (например, имена файлов и каталогов). Например, для подробного просмотра содержимого каталога documents может быть использована команда ls с ключом -l:

```
iivanova@dk4n31:~$ ls -l documents
```

В данном случае ls — это имя команды, l — ключ, documents — аргумент. Команды, ключи и аргументы должны быть отделены друг от друга пробелом. Ввод команды завершается нажатием клавиши «Enter» , после чего команда передаётся оболочке на исполнение. Результатом выполнения команды могут являться сообщения о ходе выполнения команды или об ошибках. Появление приглашения командной строки говорит о том, что выполнение команды завершено.

Иногда в GNU Linux имена программ и команд слишком длинные, однако bash может завершать имена при их вводе в терминале. Нажав клавишу «Tab» , можно завершить имя команды, программы или каталога.

Например, предположим, что нужно использовать программу `mcedit`. Для этого наберите в командной строке `mc`, затем нажмите один раз клавишу «Tab». Если ничего не происходит, то это означает, что существует несколько возможных вариантов завершения команды. Нажав клавишу «Tab» ещё раз, можно получить список имён, начинающихся с `mc`:

```
iiivanova@dk4n31:~$ mc
```

```
mc mcd mcedit mclasserase mcookie mcview
```

```
mcat mcdiff mcheck mcomp mcopy
```

```
iiivanova@dk4n31:~$ mc
```

4. Выполнение лабораторной работы.

1. Я установила Oracle VM VirtualBox на свой персональный компьютер.

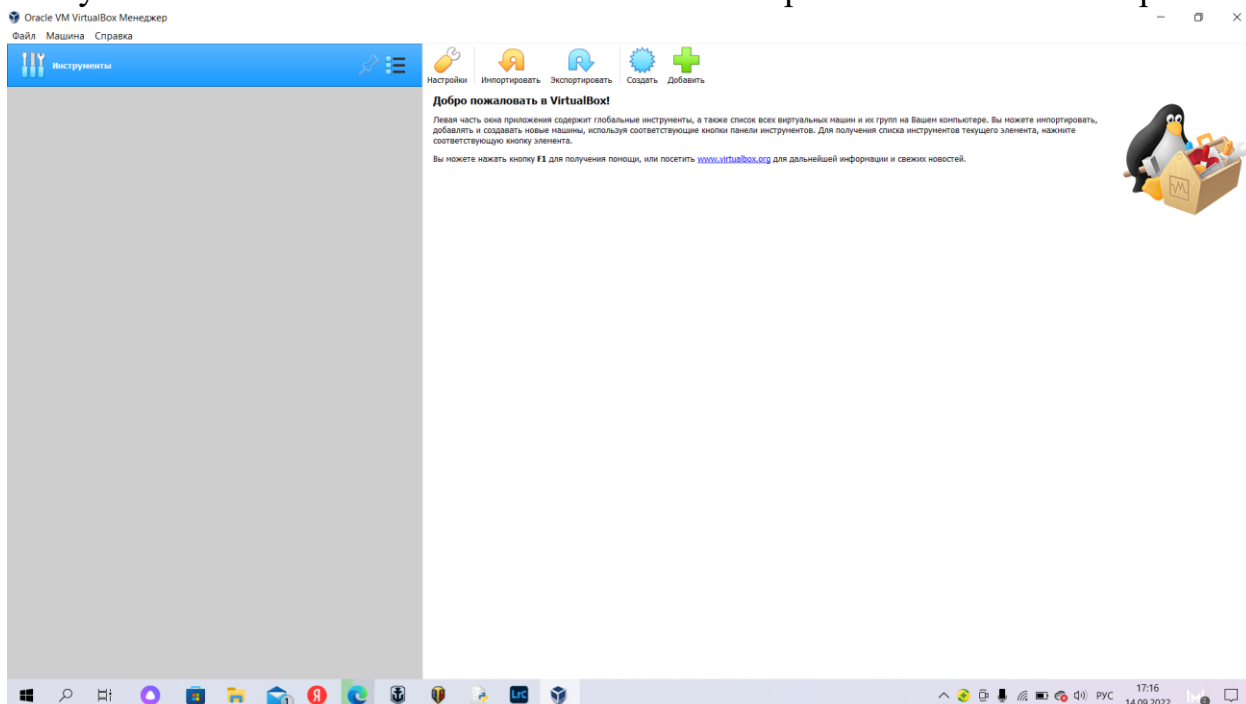


Рис. 1.1. Снимок установленной VirtualBox.

2. Создаю новую виртуальную машину с операционной системой Linux.

Вводим имя пользователь, папку для будущего хранения виртуальной машины, тип операционной системы и нужную нам версию.

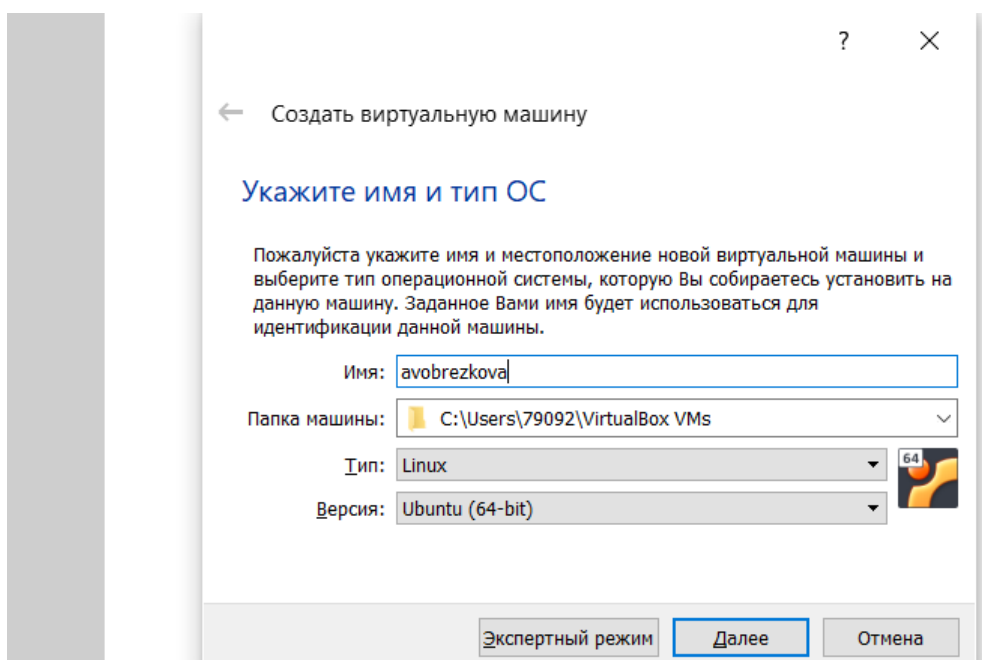


Рис. 1.2. Окно «Имя машины и тип ОС».

3. Я установила объем памяти равный 1024 МБ.

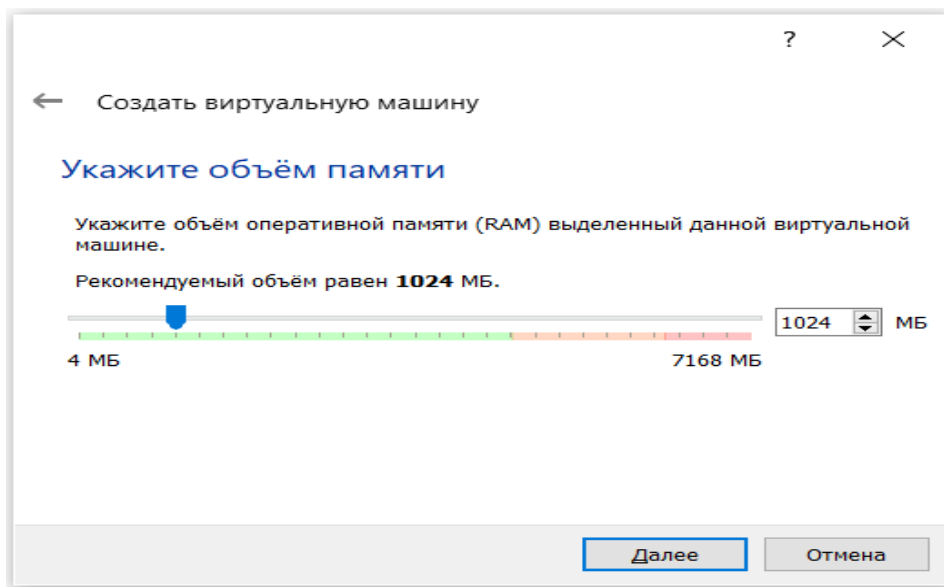


Рис. 1.3. Окно «Размер основной памяти».

4. Создала новый виртуальный жесткий диск.

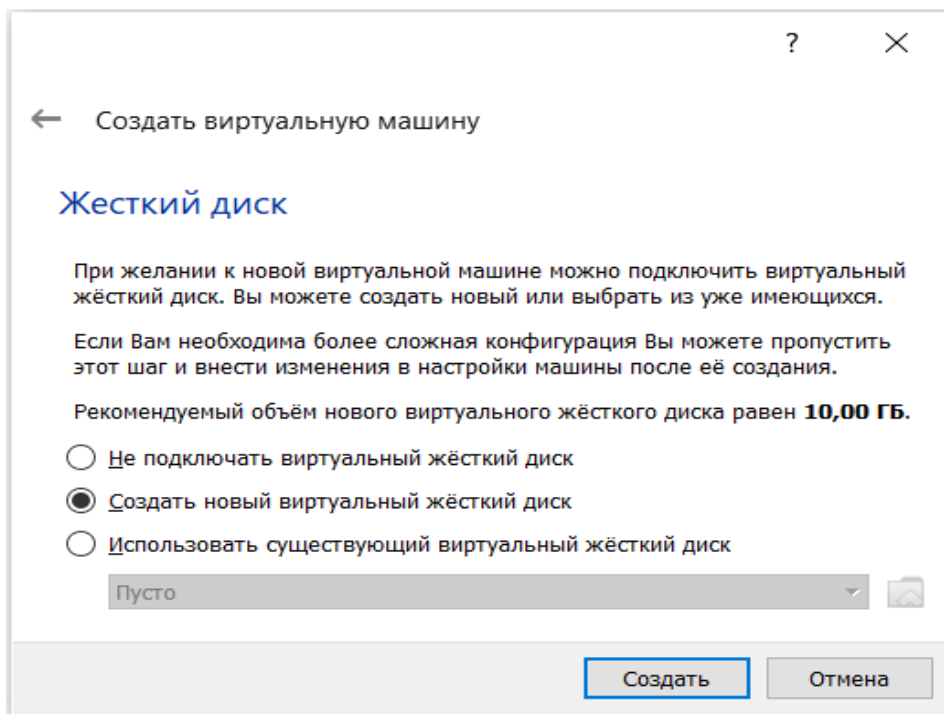


Рис. 1. 4. Окно подключения или создания жёсткого диска на виртуальной машине.

5. Выбрала тип VDI (VirtualBox Disk Image) нового виртуального жесткого диска.

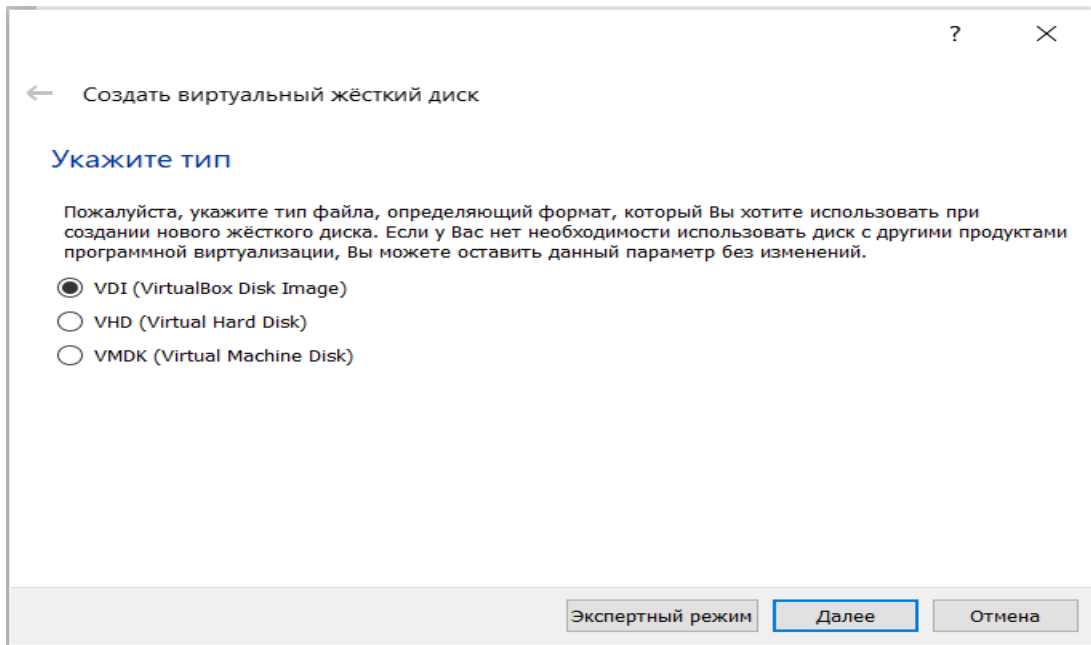


Рис. 1.5. Окно определения типа подключения виртуального жёсткого диска.

6. Указываю формат хранения (динамический виртуальный жесткий диск).

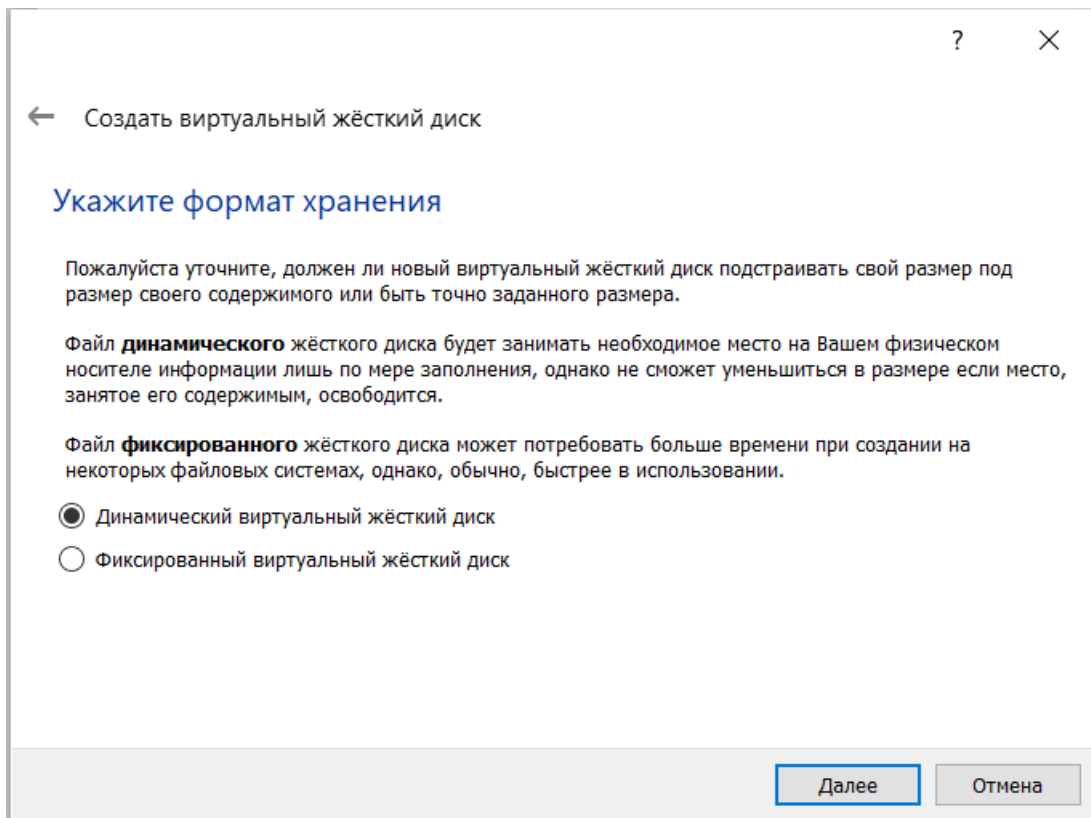


Рис. 1.6. Окно определения формата виртуального жёсткого диска.

7. Указываю имя файла, в котором будет располагаться виртуальный жесткий диска и размер этого файла.

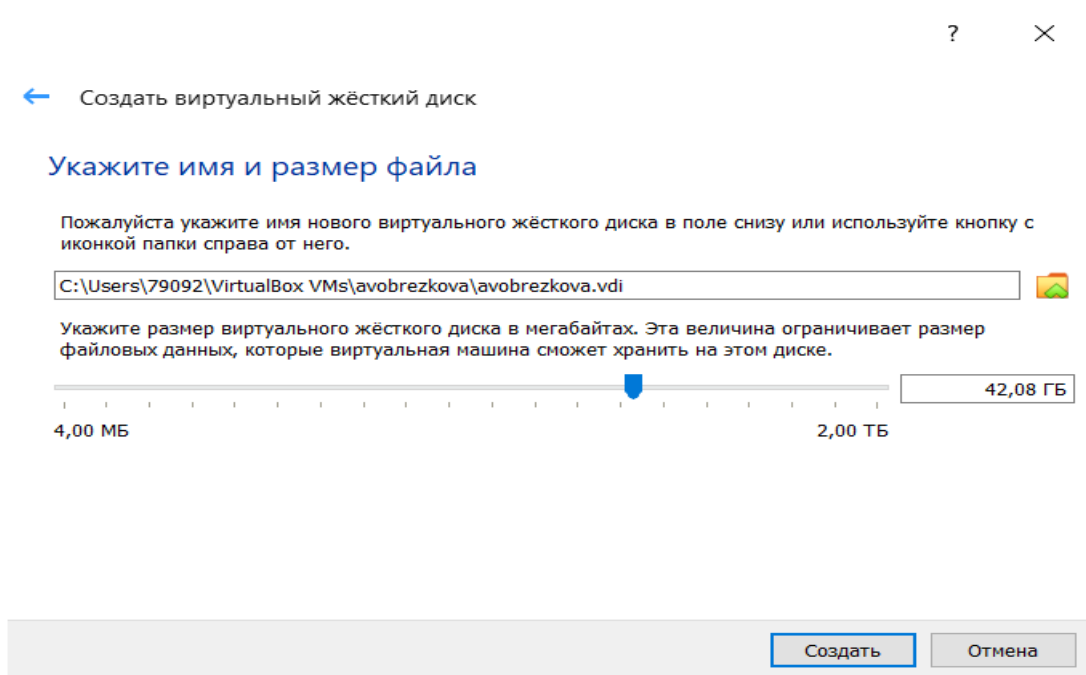


Рис. 1.7. Окно определения размера виртуального динамического жёсткого диска и его расположения.

8. После основной настройки подключила скаченный файл Ubuntu-22.04.1 в контроллер Ide.

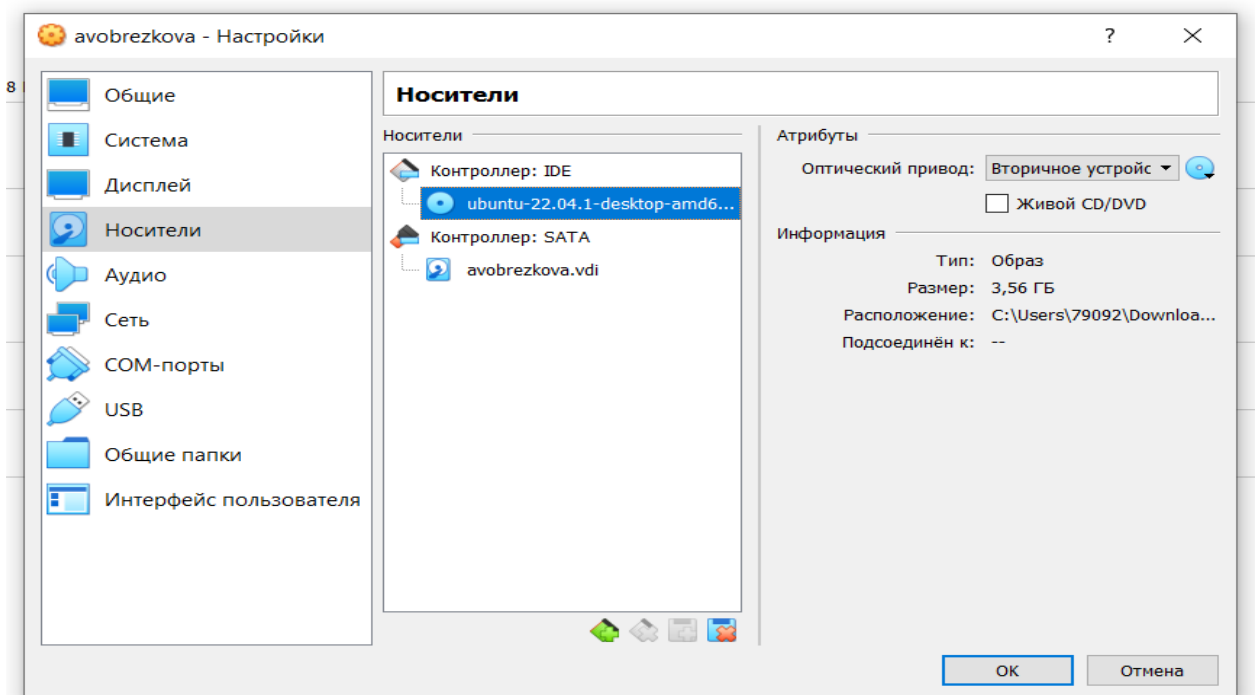


Рис. 1.8. Окно «Носители» виртуальной машины: выбор образа оптического диска.

9. Я запустила виртуальную машину и начала установку.



Рис. 1.9. Окно запуска установки образа ОС.

10. Выбрала язык виртуальной машины.

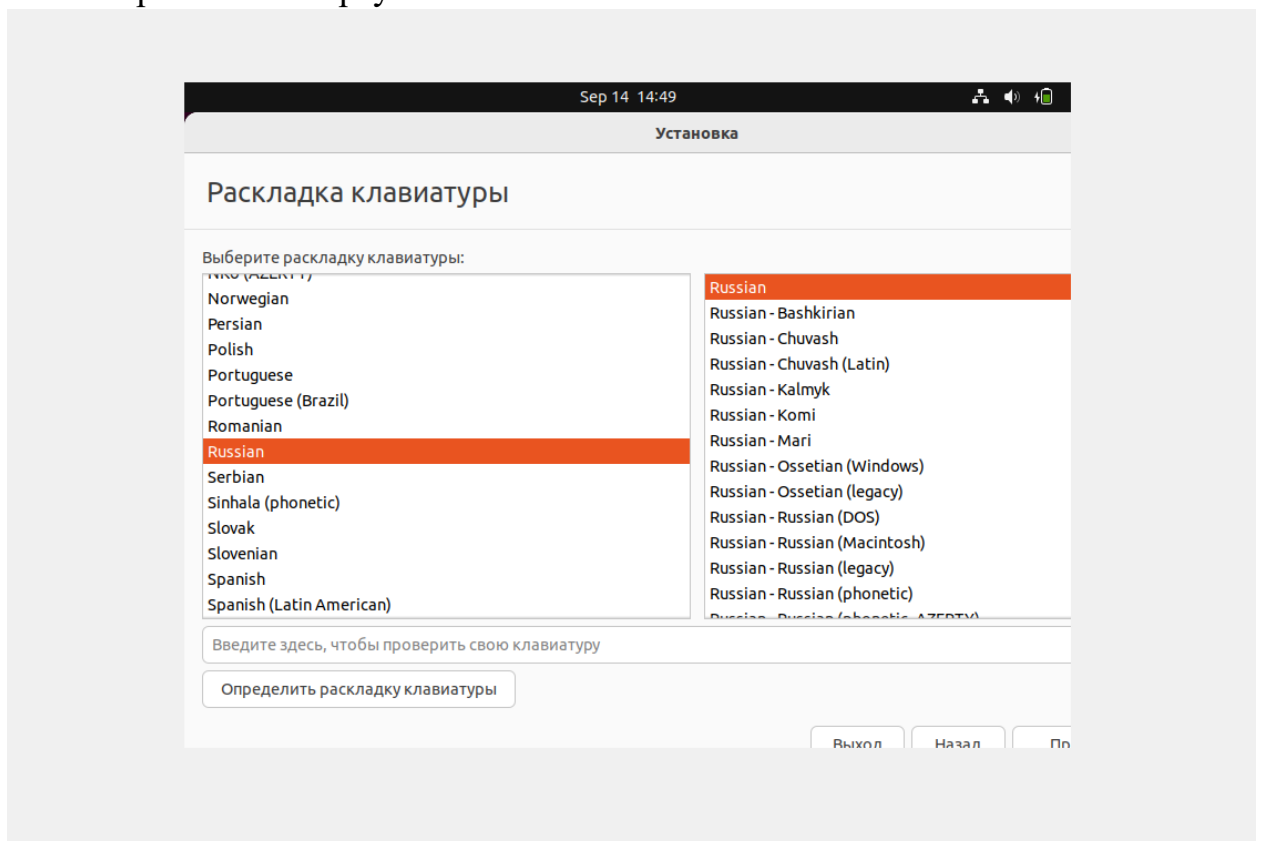


Рис. 1.10. Окно выбора языка.

11. Выбираю тип и опции установки.

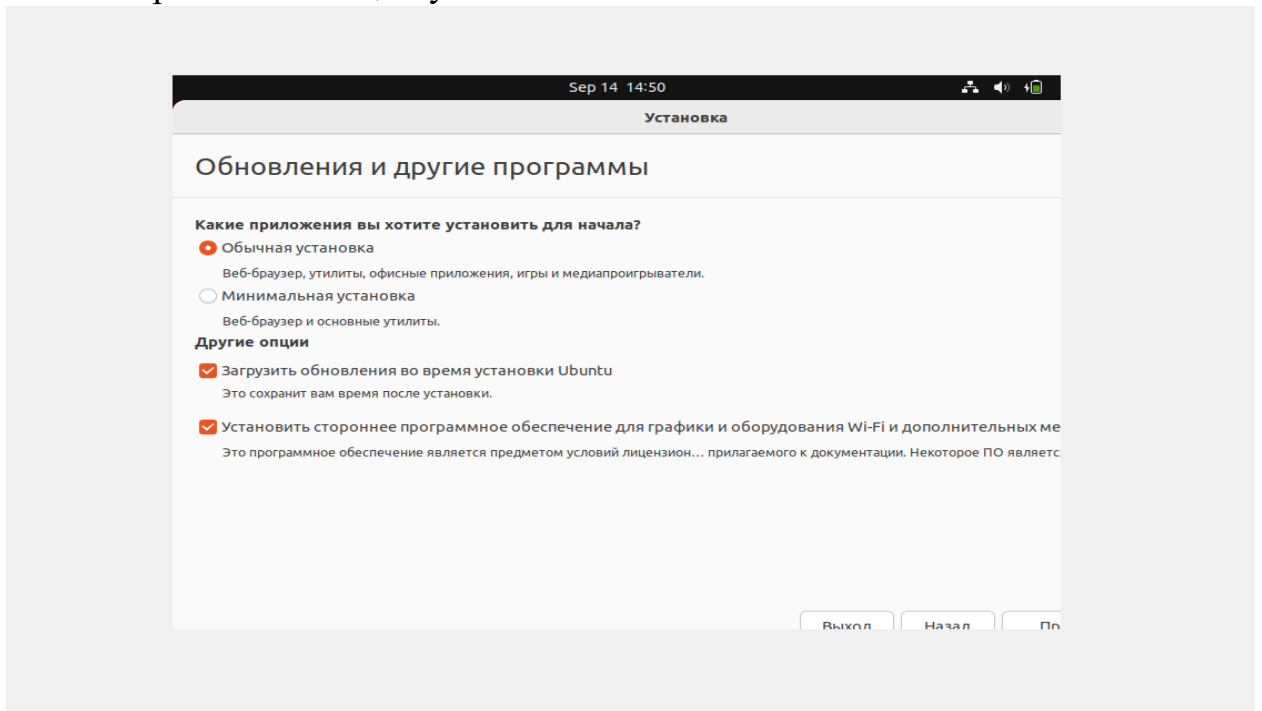


Рис. 1.11. Окно опций установки.

12. Выбираю тип установки.

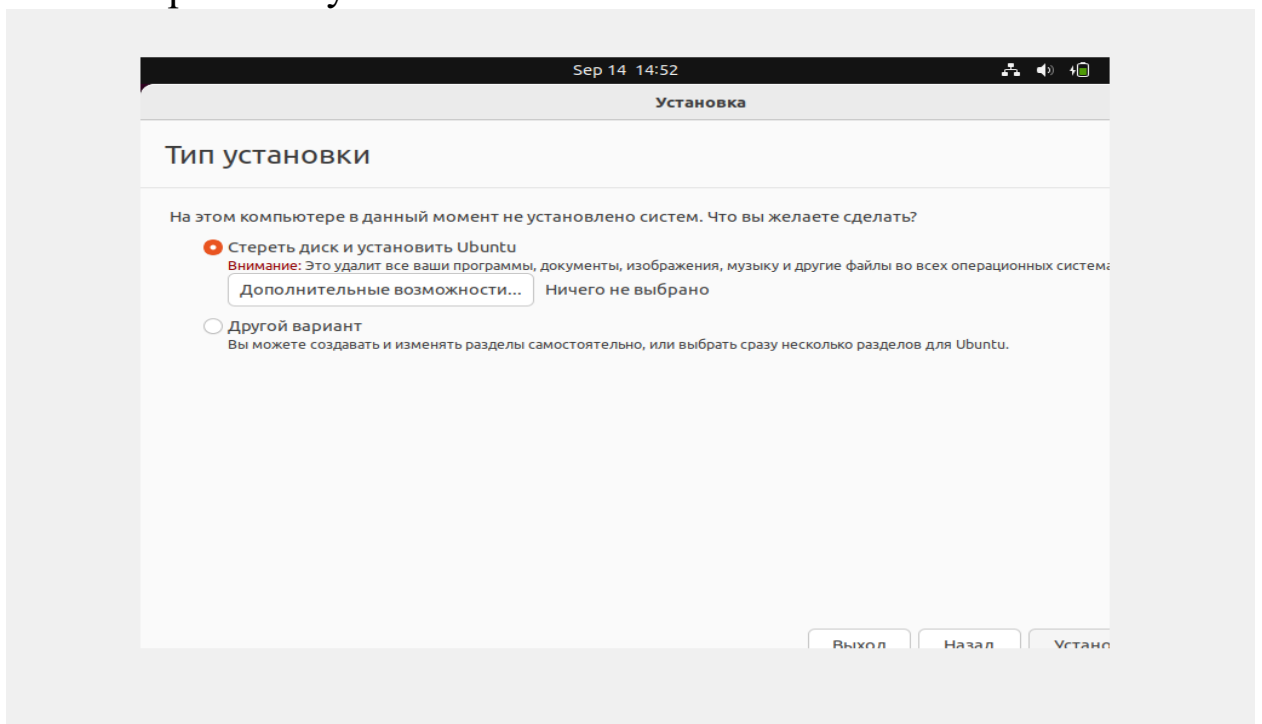


Рис. 1.12. Окно типа установки.

13. Записываю изменения на виртуальном диске.

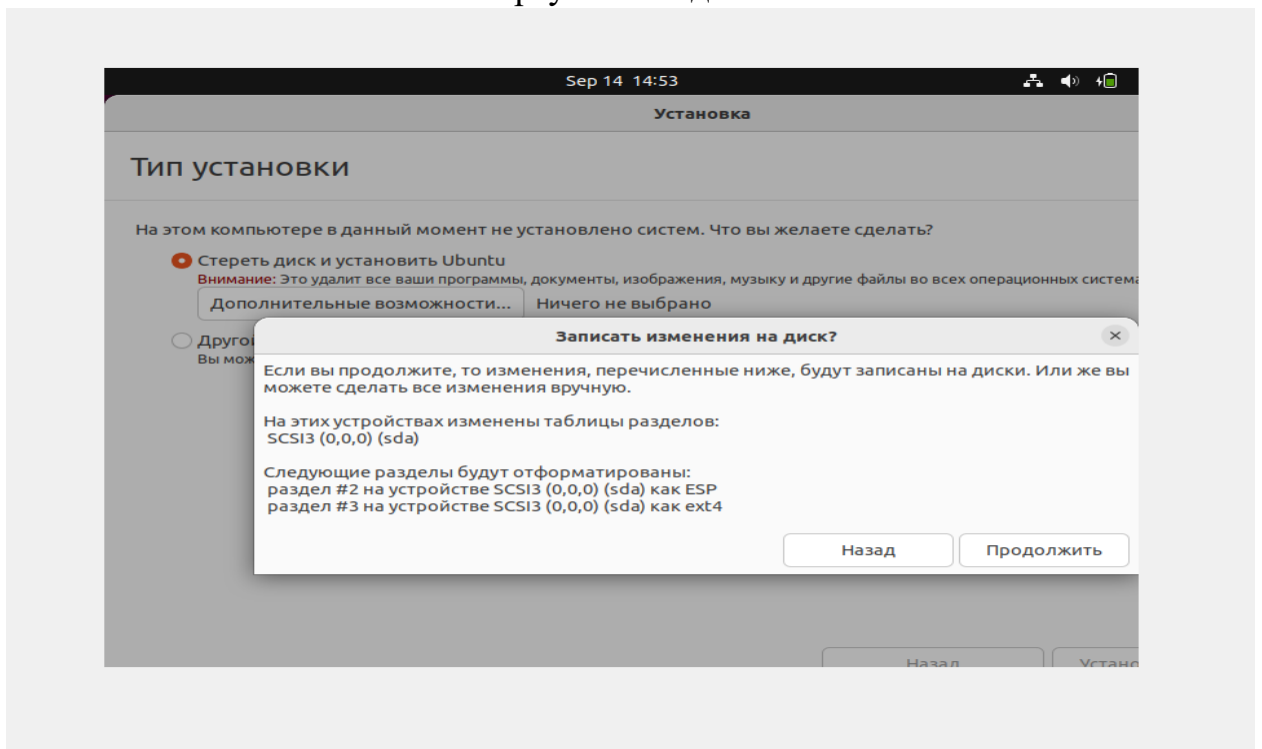


Рис. 1.13. Окно изменений диска.

14. установила регион Европа, город Москва.

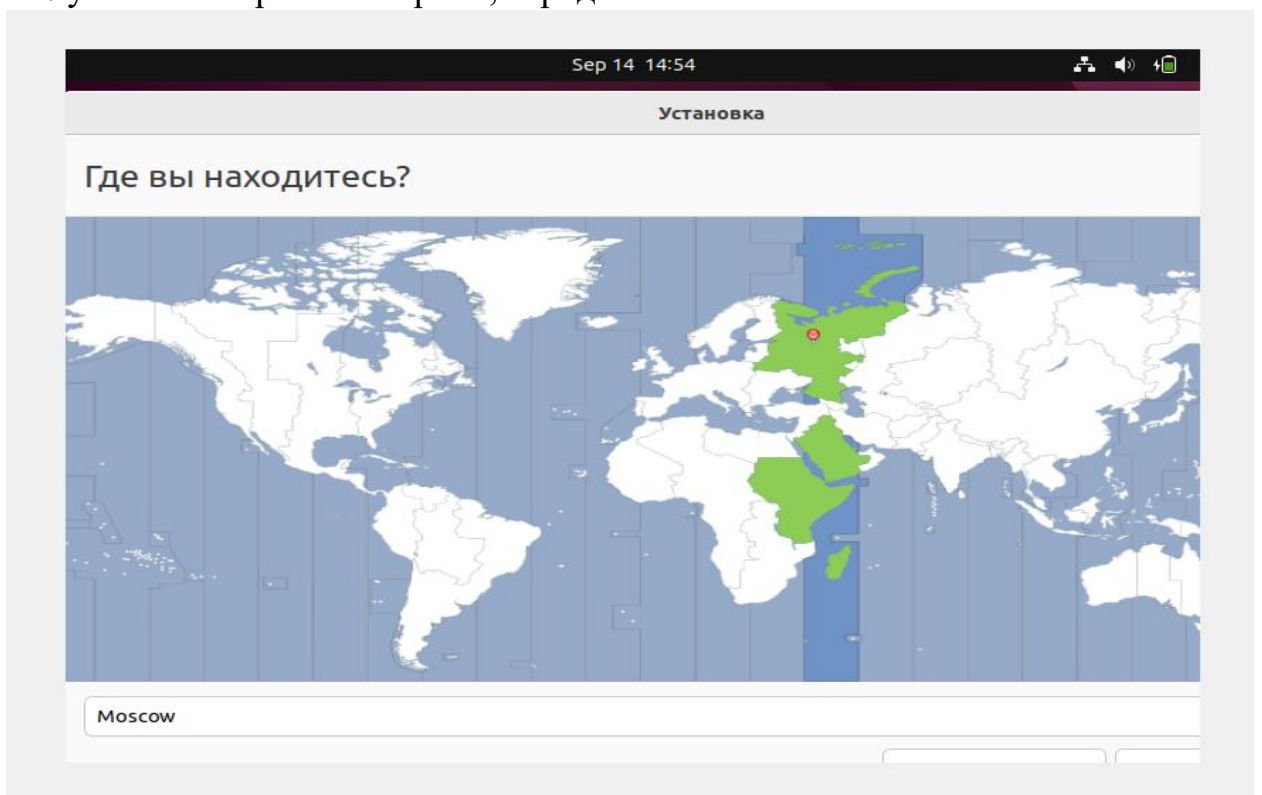


Рис. 1.14. Окно выбора часового пояса.

15. Устанавливаю имя пользователя и пароль для входа.

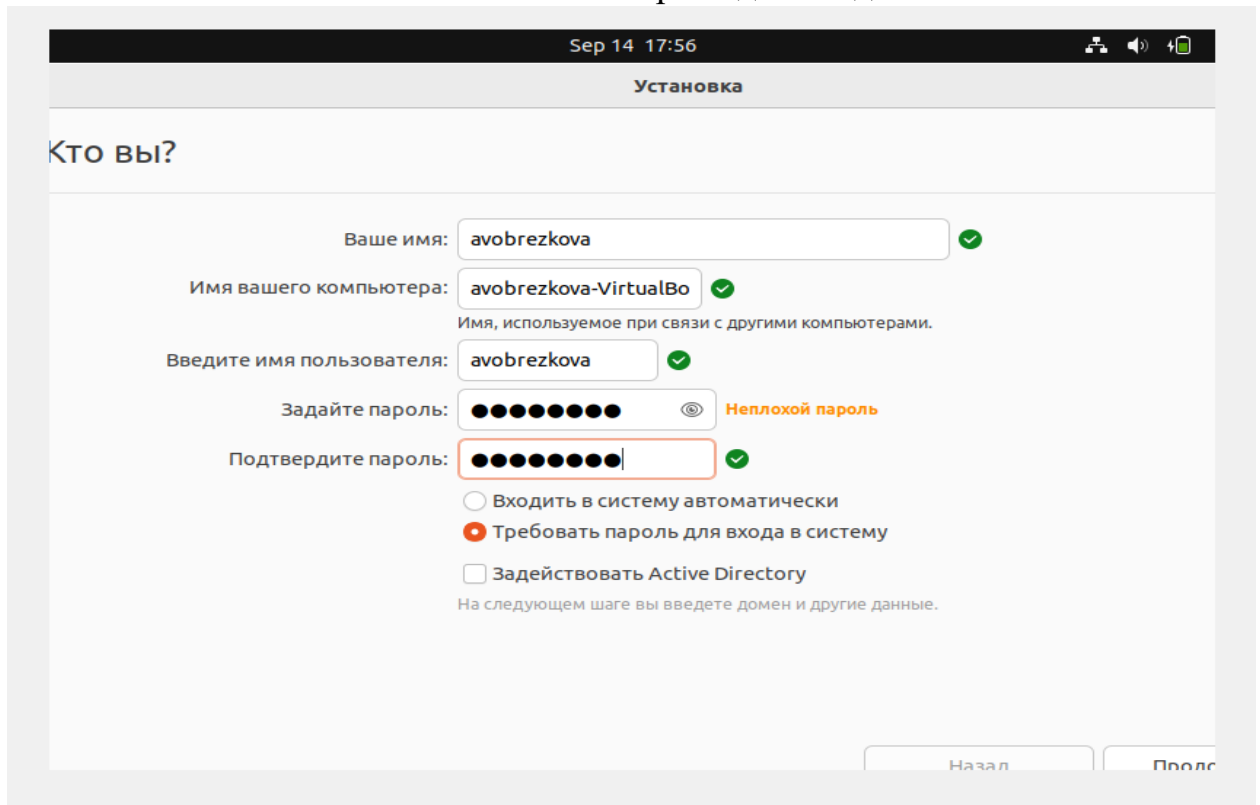


Рис. 1.15. Окно конфигурации пользователей.

16. Начала основную загрузку виртуальной машины.

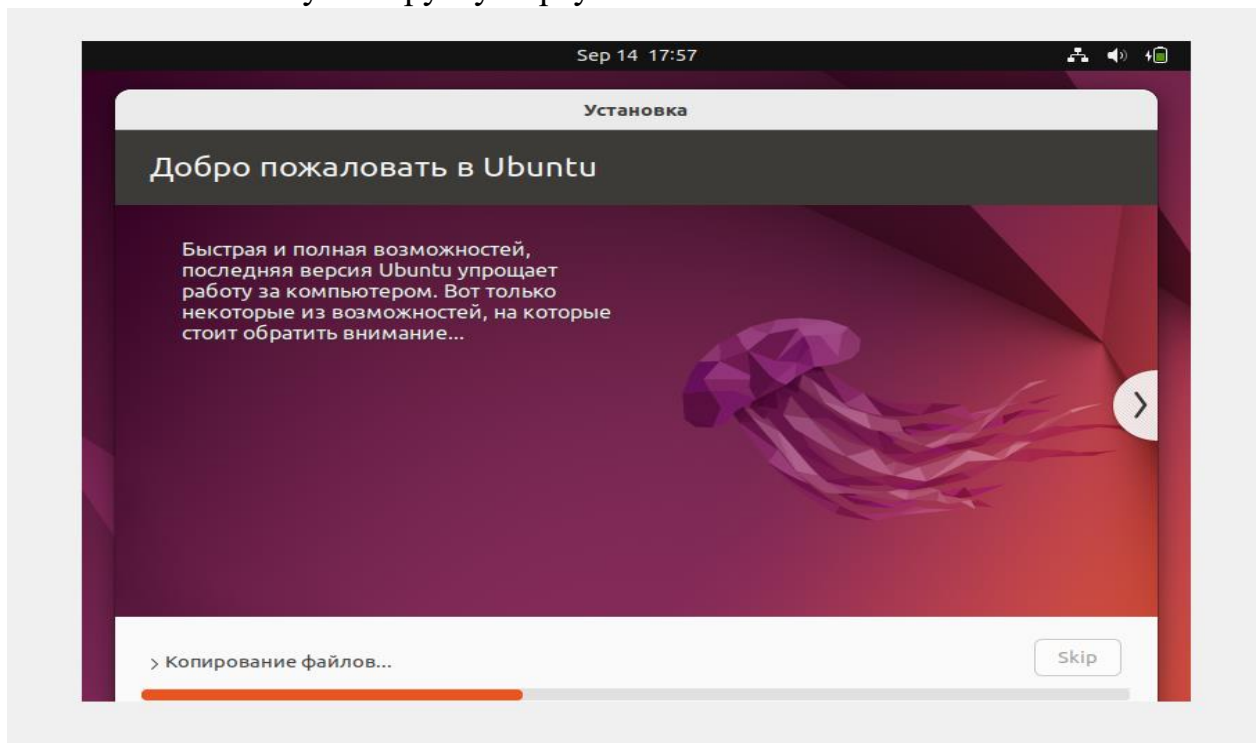


Рис. 1.16. Окно установки.

17. Закончила процесс установки.

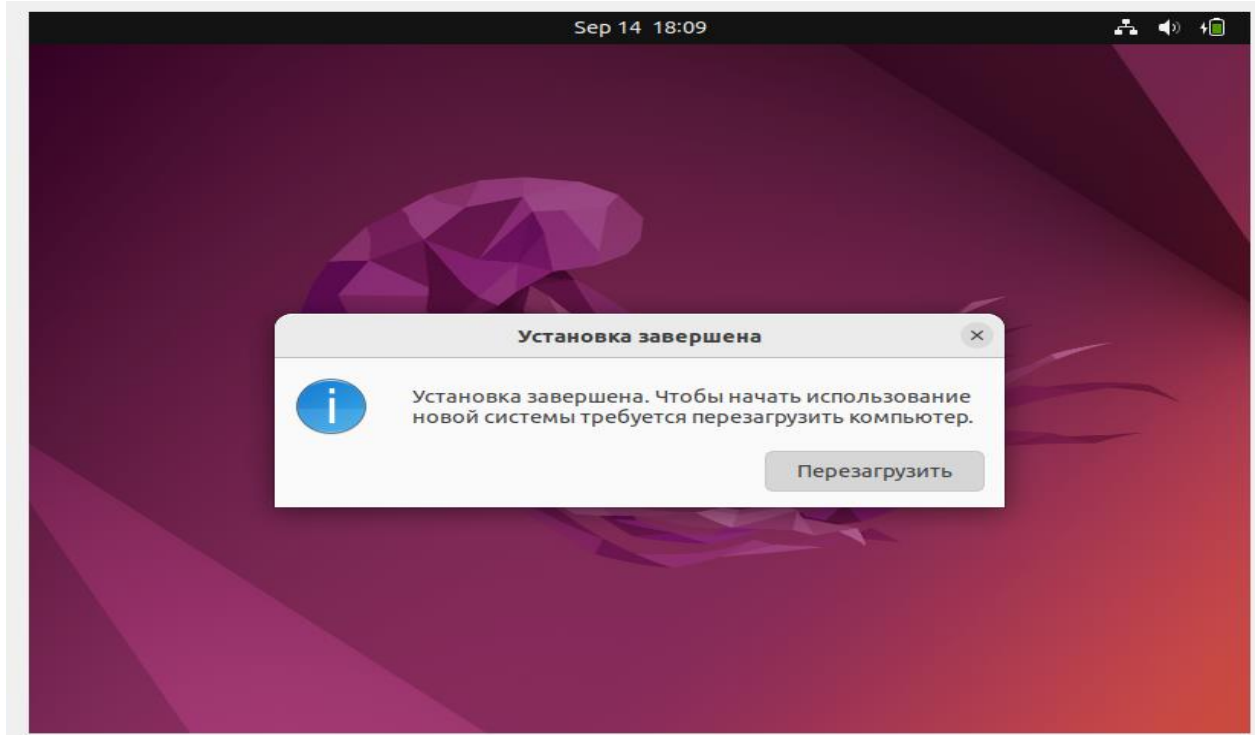


Рис. 1.17. Окно завершения установки.

18. После установки выхожу в основные настройки виртуальной машины и изымаю жесткий диск, заново запускаю виртуальную машину.

19. Выполнила основной вход в виртуальную машину.

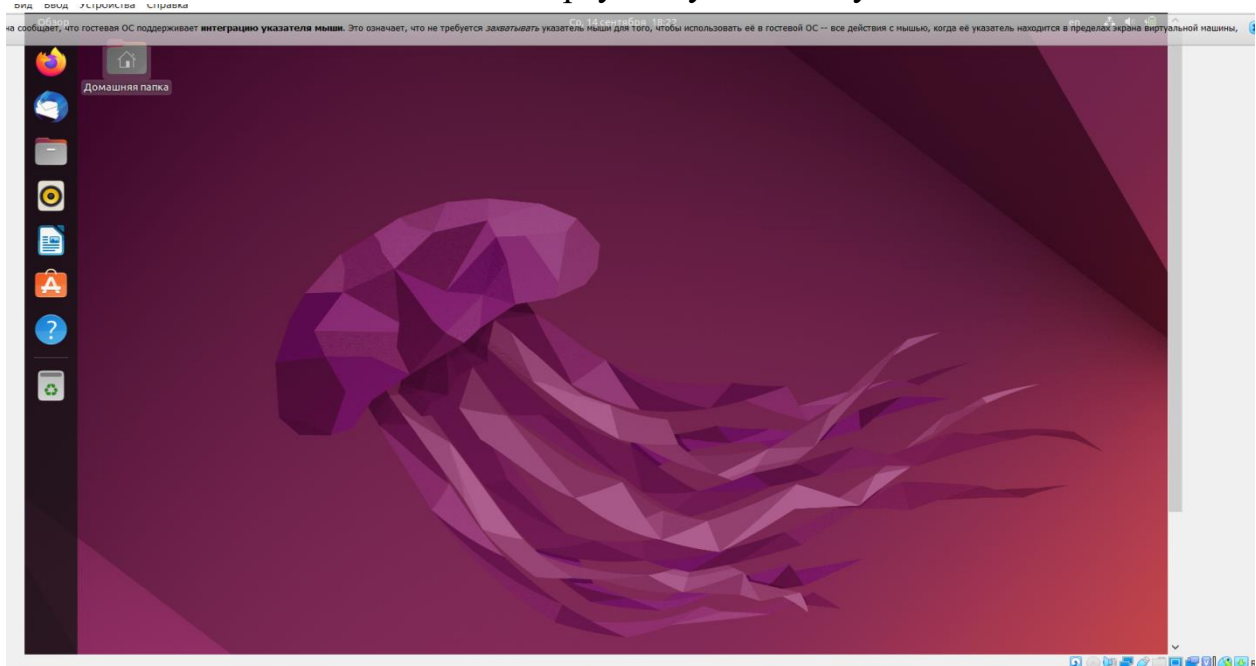


Рис. 1.18. Окно «Рабочий стол виртуальной машины nV операционной системе Linux».

Дальнейшие действия будут производиться на виртуальной машине на основе Fedora, по причине того, что скаченная мною машина на основе Ubuntu перестала работать и открываться на моем компьютере.

5. Самостоятельная работа.

1. Запустила установленную в VirtualBox Fedora операционную систему Linux.

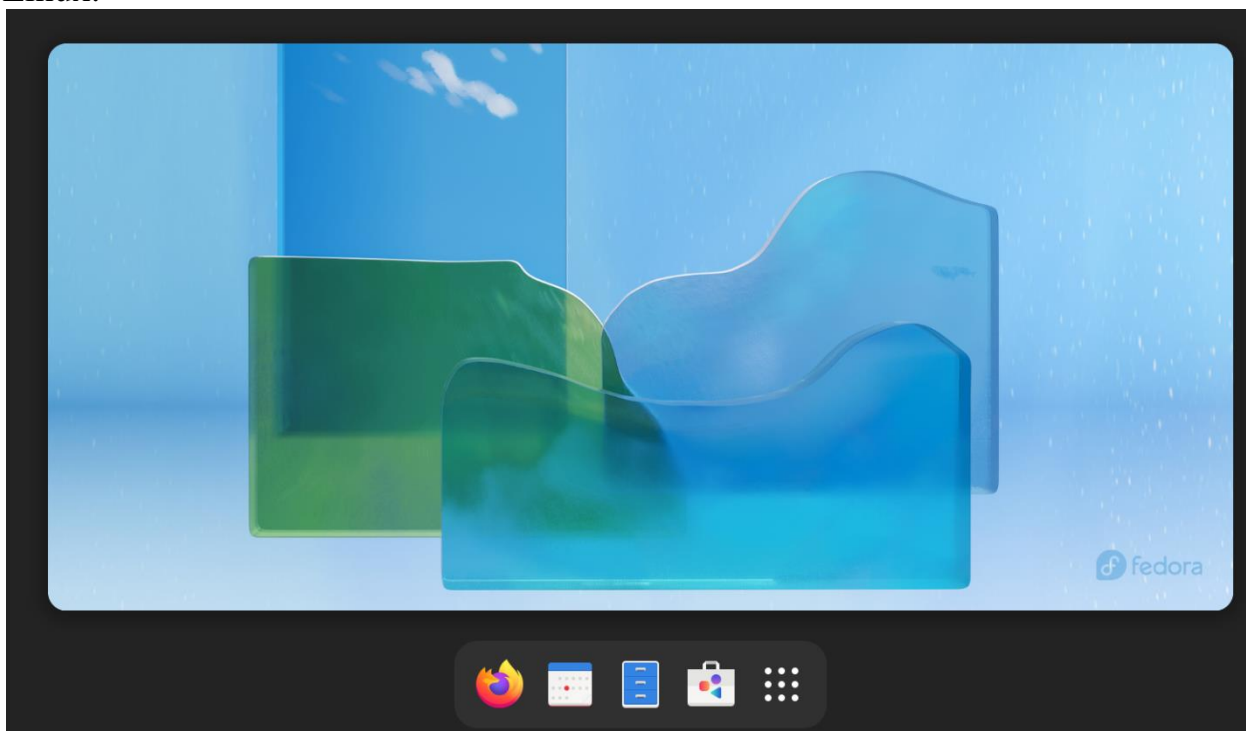


Рис. 2.1. Окно «Главный экран Linux»

2. Запустила браузер Firefox.

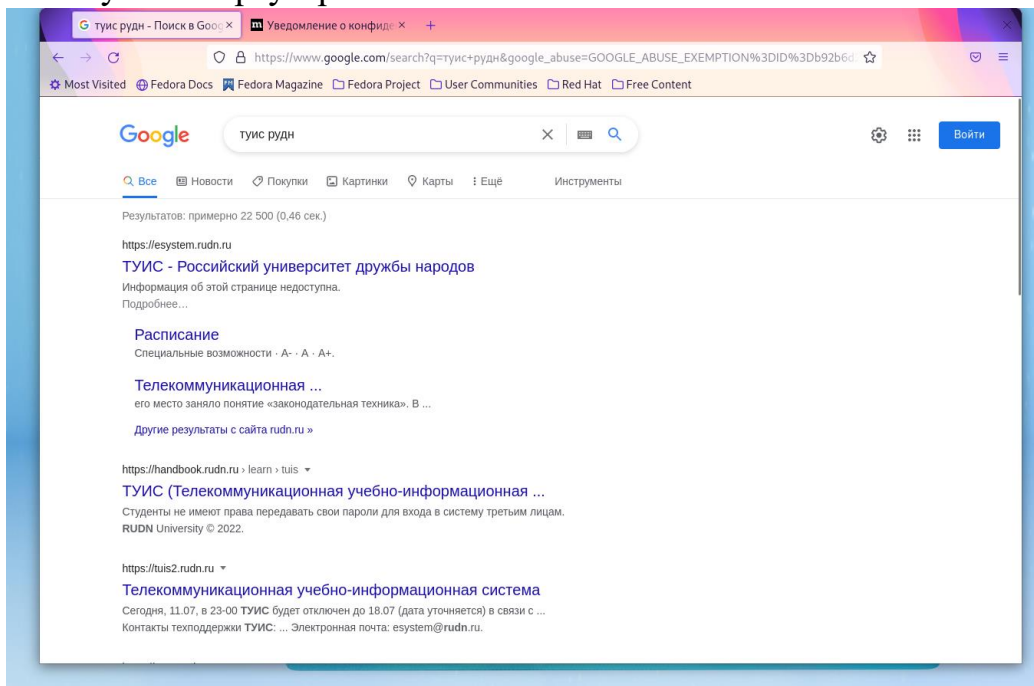


Рис. 2.2. Окно браузера.

3. Запустила текстовый процессор LibreOffice Writer.

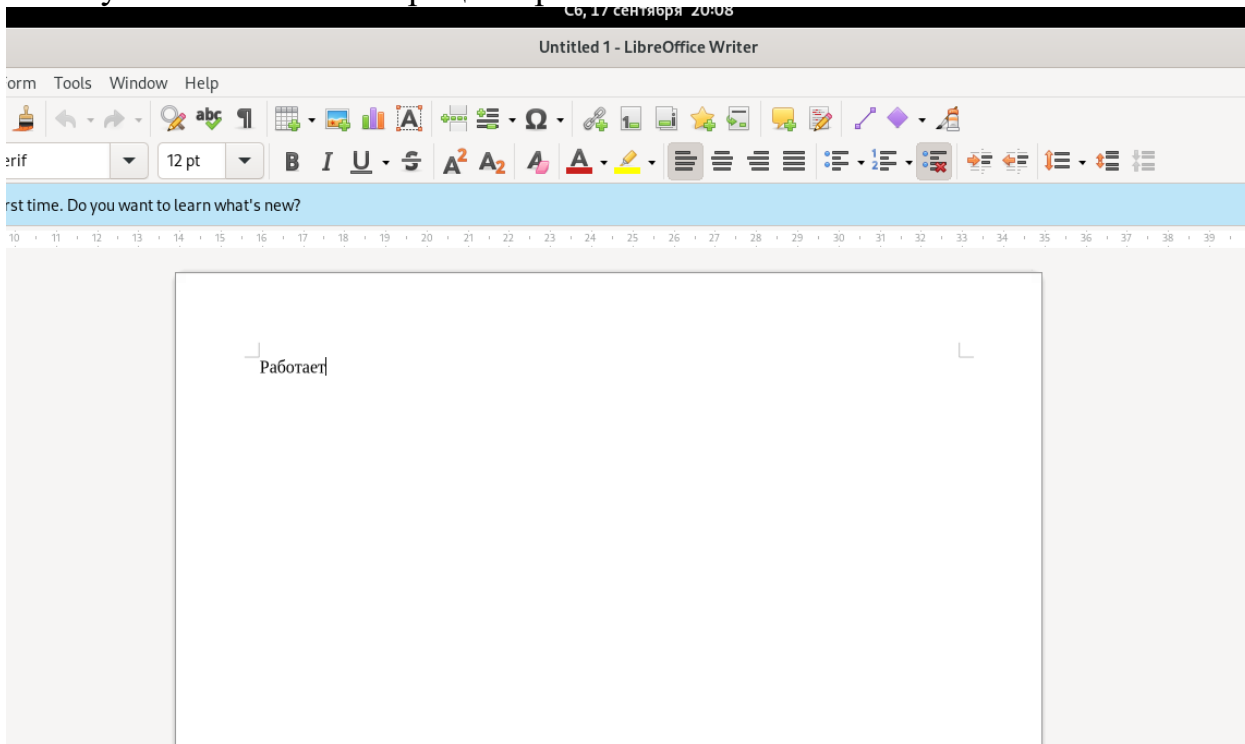


Рис. 2.3. Окно текстового процессора.

4. Запустила текстовый редактор.

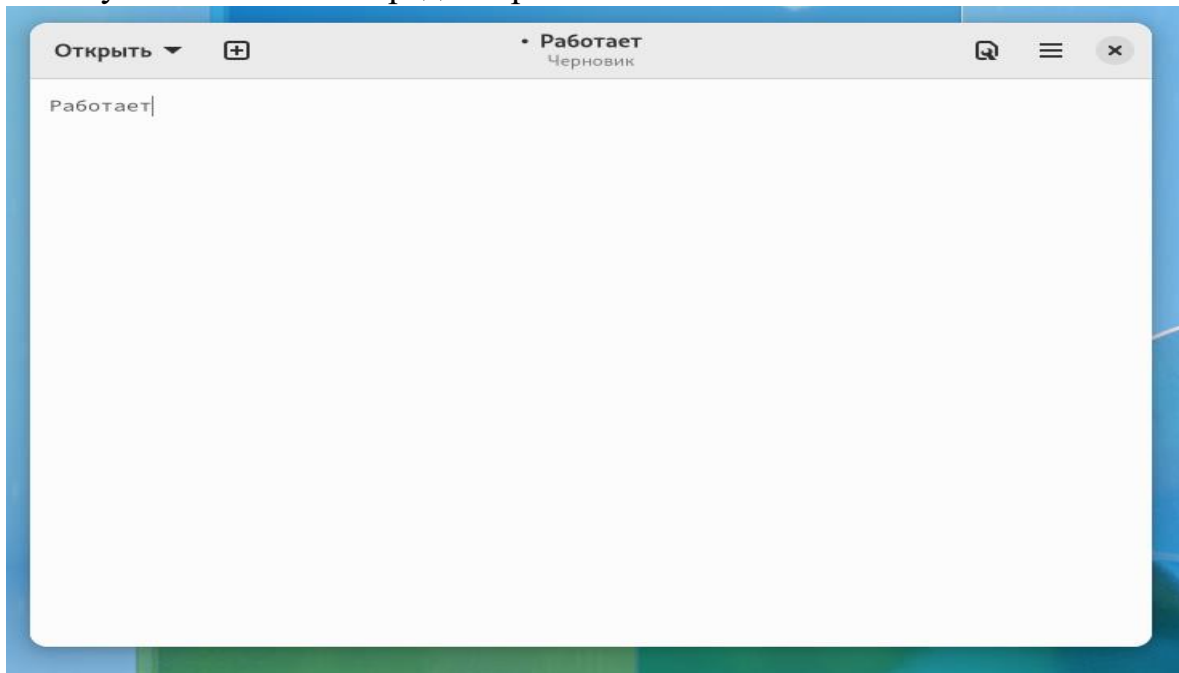


Рис. 2.4. Окно текстового редактора.

5. С помощью терминала установила Midnight Commander (mc), так же через команду «mc» проверила его работу.

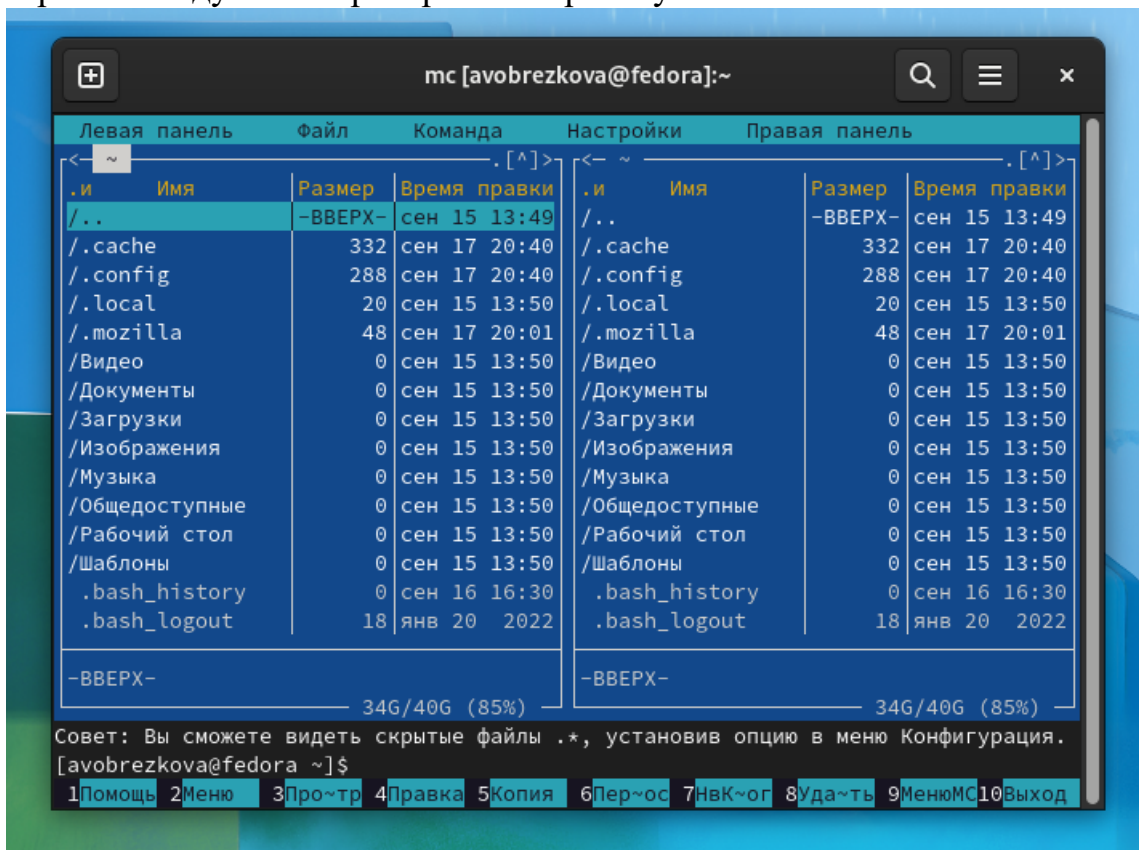
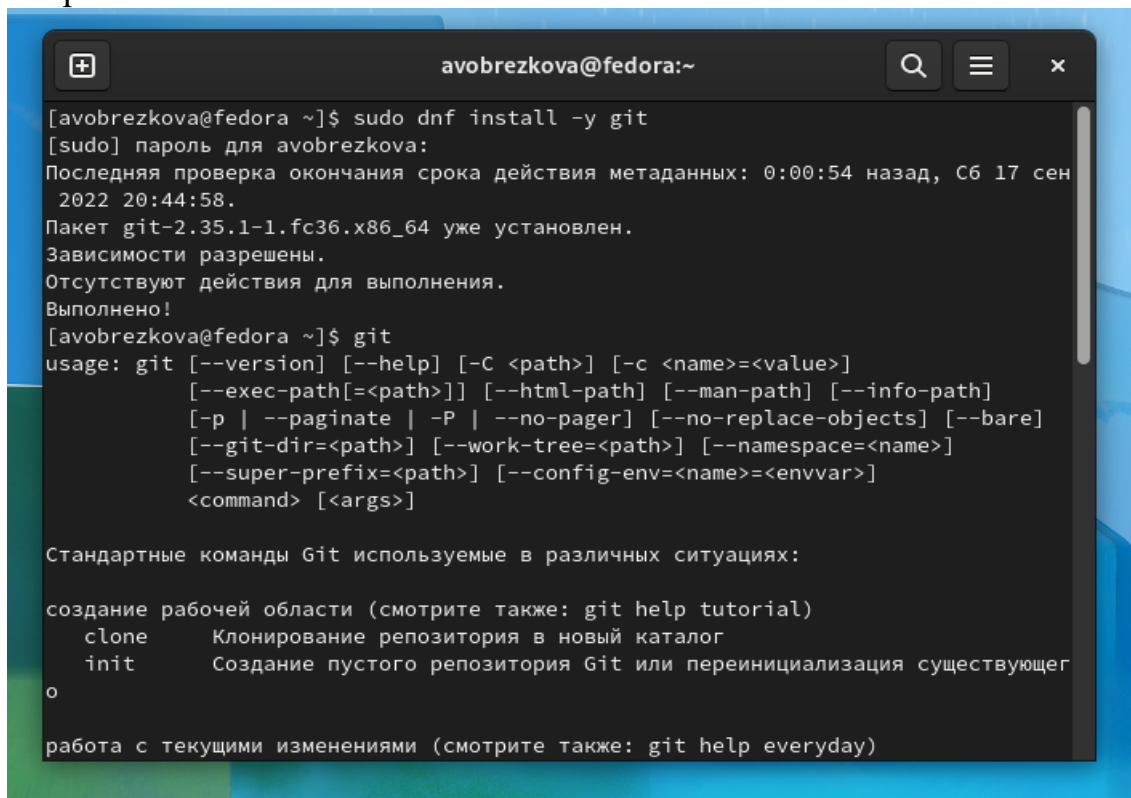


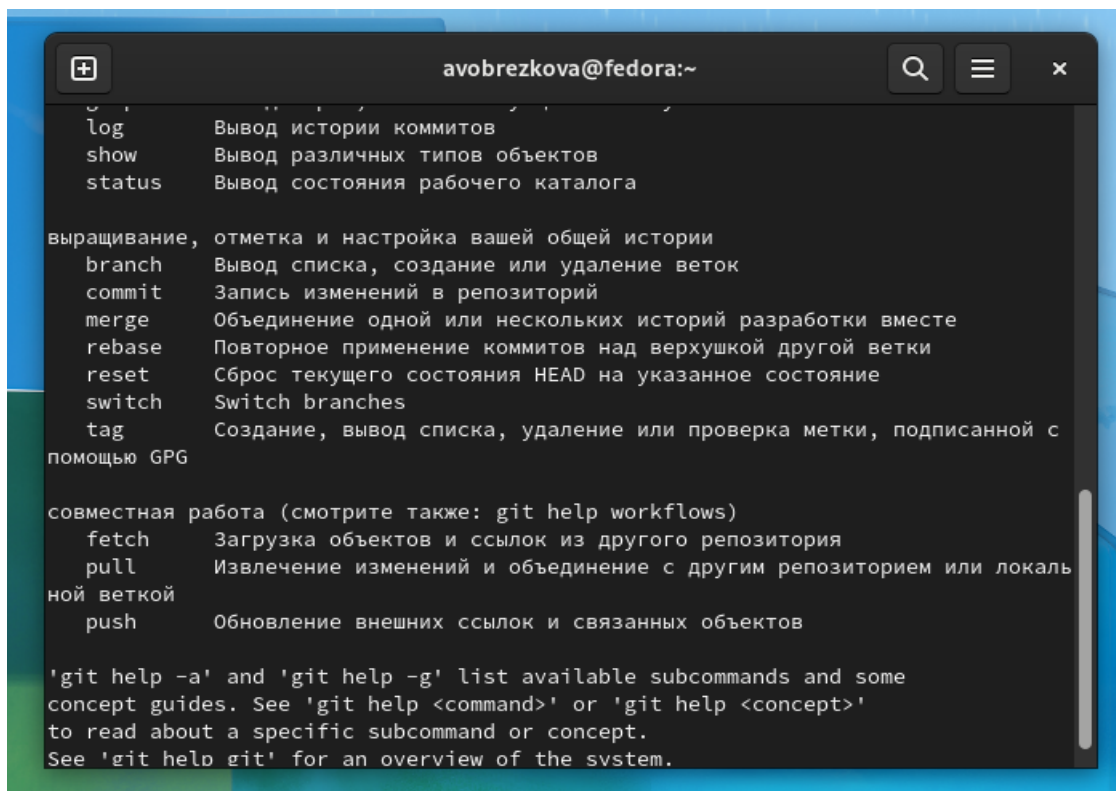
Рис. 2.5. Подключение Midnight Commander.

6. Проверила наличие системы Git. С помощью команды «git» вывела данные в терминале.



```
avobrezkova@fedora:~  
[avobrezkova@fedora ~]$ sudo dnf install -y git  
[sudo] пароль для avobrezkova:  
Последняя проверка окончания срока действия метаданных: 0:00:54 назад, Сб 17 сен 2022 20:44:58.  
Пакет git-2.35.1-1.fc36.x86_64 уже установлен.  
Зависимости разрешены.  
Отсутствуют действия для выполнения.  
Выполнено!  
[avobrezkova@fedora ~]$ git  
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]  
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]  
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]  
      [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]  
      [--super-prefix=<path>] [--config-env=<name>=<envvar>]  
      <command> [<args>]  
  
Стандартные команды Git используемые в различных ситуациях:  
  
создание рабочей области (смотрите также: git help tutorial)  
  clone      Клонирование репозитория в новый каталог  
  init       Создание пустого репозитория Git или переинициализация существующего  
о  
  
работа с текущими изменениями (смотрите также: git help everyday)
```

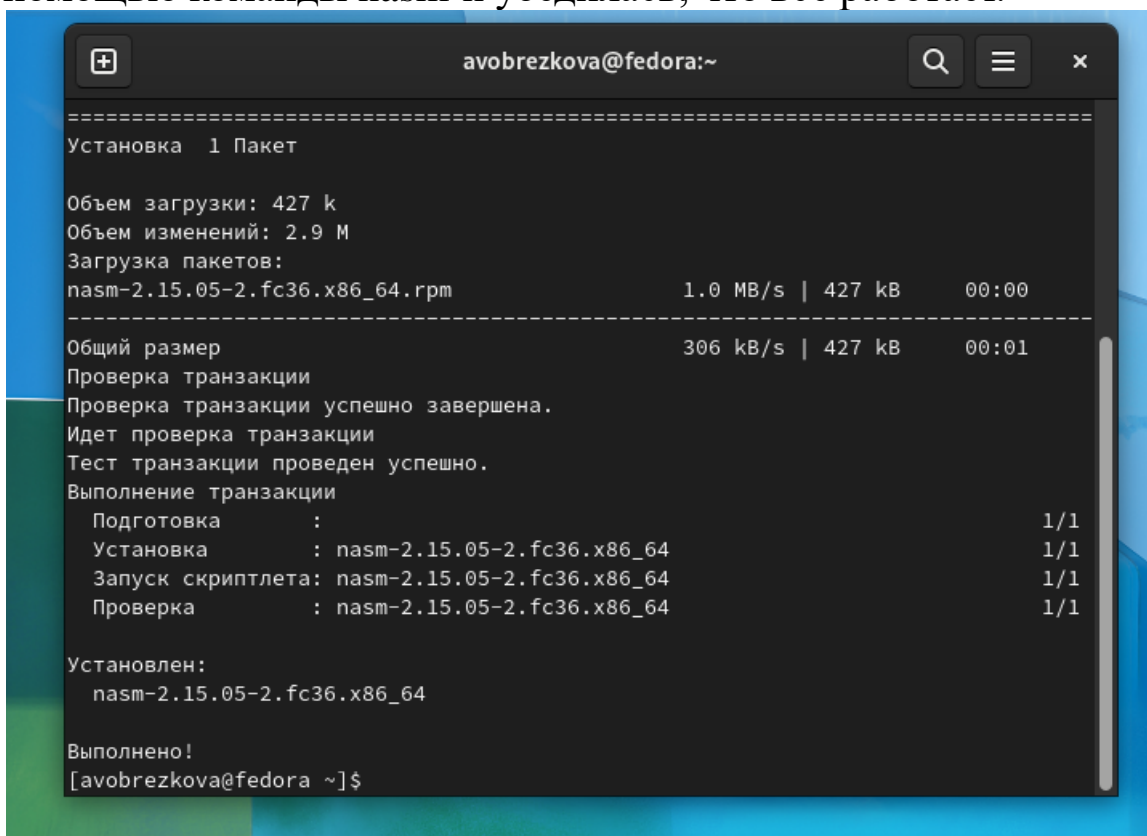
Рис. 2.6. Подключение Git.



```
avobrezkova@fedora:~  
log      Вывод истории коммитов  
show     Вывод различных типов объектов  
status   Вывод состояния рабочего каталога  
  
выращивание, отметка и настройка вашей общей истории  
branch   Вывод списка, создание или удаление веток  
commit   Запись изменений в репозиторий  
merge    Объединение одной или нескольких историй разработки вместе  
rebase   Повторное применение коммитов над верхушкой другой ветки  
reset    Сброс текущего состояния HEAD на указанное состояние  
switch   Switch branches  
tag      Создание, вывод списка, удаление или проверка метки, подписанной с помощью GPG  
  
совместная работа (смотрите также: git help workflows)  
fetch    Загрузка объектов и ссылок из другого репозитория  
pull     Извлечение изменений и объединение с другим репозиторием или локальной веткой  
push     Обновление внешних ссылок и связанных объектов  
  
'git help -a' and 'git help -g' list available subcommands and some concept guides. See 'git help <command>' or 'git help <concept>' to read about a specific subcommand or concept.  
See 'git help git' for an overview of the system.
```

Рис. 2.7. Подключение Git.

7. Я проверила наличие системы Nasm (Netwide assembler). С помощью команды `nasm-h` убедилась, что все работает.



```
=====
Установка 1 Пакет

Объем загрузки: 427 k
Объем изменений: 2.9 M
Загрузка пакетов:
nasm-2.15.05-2.fc36.x86_64.rpm                1.0 MB/s | 427 kB    00:00
=====
Общий размер                                306 kB/s | 427 kB    00:01
Проверка транзакции
Проверка транзакции успешно завершена.
Идет проверка транзакции
Тест транзакции проведен успешно.
Выполнение транзакции
Подготовка      :                               1/1
Установка       : nasm-2.15.05-2.fc36.x86_64  1/1
Запуск скрипта  : nasm-2.15.05-2.fc36.x86_64  1/1
Проверка        : nasm-2.15.05-2.fc36.x86_64  1/1

Установлен:
nasm-2.15.05-2.fc36.x86_64

Выполнено!
[avobrezkova@fedora ~]$
```

Рис. 2.8. Подключение Nasm.

6. Вывод.

Я приобрела практические навыки по установке операционной системы Linux на виртуальную машину и настроила необходимые для дальнейшей эксплуатации сервисы.

7. Список литературы.

1. <https://www.virtualbox.org/> - виртуальная машина
2. <https://getfedora.org/> - операционная система
3. [Шаблон отчёта по лабораторной работе \(rudn.ru\)](#) – шаблон по оформлению
4. [Архитектура ЭВМ \(rudn.ru\)](#) – инструкция по выполнению работы

8. Список иллюстраций.

Рис. 1.1. Снимок установленной VirtualBox.	7
Рис. 1.2. Окно «Имя машины и тип ОС».....	7
Рис. 1.3. Окно «Размер основной памяти».....	8
Рис. 1. 4. Окно подключения или создания жёсткого диска на виртуальной машине.....	8
Рис. 1.5. Окно определения типа подключения виртуального жёсткого диска.	9
Рис. 1.6. Окно определения формата виртуального жёсткого диска.	9
Рис. 1.7. Окно определения размера виртуального динамического жёсткого диска и его расположения.	10
8. После основной настройки подключила скаченный файл Ubuntu-22.04.1 в контроллер Ide.	10
Рис. 1.8. Окно «Носители» виртуальной машины: выбор образа оптического диска.	10
Рис. 1.10. Окно выбора языка.	11
Рис. 1.11. Окно опций установки.....	12
Рис. 1.12. Окно типа установки.	12
Рис. 1.13. Окно изменений диска.	13
Рис. 1.14. Окно выбора часового пояса.	13
Рис. 1.15. Окно конфигурации пользователей.	14
Рис. 1.16. Окно установки.	14
Рис. 1.17. Окно завершения установки.	15
Рис. 1.18. Окно «Рабочий стол виртуальной машины nB операционной системе Linux». ..	15
Рис. 2.1. Окно «Главный экран Linux».....	16
Рис. 2.2. Окно браузера.	17
Рис. 2.3. Окно текстового процессора.	17
Рис. 2.4. Окно текстового редактора.	18
Рис. 2.5. Подключение Midnight Commander.	18
Рис. 2.6. Подключение Git.	19
Рис. 2.7. Подключение Git.	19
Рис. 2.8. Подключение Nasm.	20