

# **Лабораторная работа №1**

**Дисциплина: операционные системы**

**Пронякова Ольга Максимовна**

# Содержание

<b>1 Цель работы</b>	<b>5</b>
<b>2 Задание</b>	<b>6</b>
<b>3 Теоретическое введение</b>	<b>7</b>
3.1 Введение в GNU Linux . . . . .	7
3.2 Введение в командную строку GNU Linux . . . . .	8
<b>4 Выполнение лабораторной работы</b>	<b>10</b>
4.1 Настройка VirtualBox . . . . .	10
4.2 Запуск виртуальной машины и установка системы . . . . .	13
4.3 Завершение установки . . . . .	17
<b>5 Задания для самостоятельной работы</b>	<b>20</b>
<b>6 Выполнение домашней работы</b>	<b>24</b>
<b>7 Ответы на контрольные вопросы</b>	<b>30</b>
<b>8 Выводы</b>	<b>32</b>
<b>Список литературы</b>	<b>33</b>

# Список иллюстраций

4.1	Окно «Имя машины и тип ОС	11
4.2	Настройка виртуальной машины	12
4.3	Окно «Носители» виртуальной машины: выбор образа оптического диска	13
4.4	Окно запуска установки образа ОС	14
4.5	Окно выбора языка	15
4.6	Окно создания пользователя и задания пароля	16
4.7	Окно Авторизация	17
4.8	Выключение системы	18
4.9	Извлечение образа диска	19
5.1	Браузер Firefox	20
5.2	Установила основное программное обеспечение Midnight Commander (mc)	21
5.3	Установила основное программное обеспечение Git	22
5.4	Установила основное программное обеспечение Nasm (Netwide Assembler)	23
6.1	Анализ последовательности загрузки системы	24
6.2	Анализ последовательности загрузки системы	25
6.3	Версия ядра Linux, Частота процессора, Модель процессора	26
6.4	Объём доступной оперативной памяти	26
6.5	Объём доступной оперативной памяти	27
6.6	Тип обнаруженного гипервизора	28
6.7	Тип файловой системы корневого раздела	29
6.8	Последовательность монтирования файловых систем	29

# **Список таблиц**

# **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**

1. Настройка VirtualBox
2. Запуск виртуальной машины и установка системы
3. Завершение установки
4. Задания для самостоятельной работы
5. Выполнение домашней работы
6. Ответы на контрольные вопросы

# **3 Теоретическое введение**

## **3.1 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).

## 3.2 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 пользователю 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 ), : *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:  
iivanova@dk4n31:~\$ mc mc mcd mcedit mclasserase mcookie mcview mcat mcdiff  
mcheck mcomp mcopy iivanova@dk4n31:~\$ mc

# **4 Выполнение лабораторной работы**

## **4.1 Настройка VirtualBox**

Загрузила в дисплейном классе операционную систему Linux. Вошла в систему. Запустила терминал (через главное меню Приложения Стандартные Терминал (или Консоль) или нажав Ctrl + Alt + t ) и перешла в каталог /var/tmp cd /var/tmp Создала каталог с моим именем пользователя. Проверила в свойствах VirtualBox месторасположение каталога для виртуальных машин. Для этого в VirtualBox выбрала Файл Свойства , вкладка Общие. Также сменила комбинацию для хост-клавиши, которая используется для освобождения курсора мыши, который может захватить виртуальная машина. Файл Свойства , вкладка Ввод Виртуальная машина. Создала новую виртуальную машину. Для этого в VirtualBox выбрала Машина Создать . Указала имя виртуальной машины (мой логин в дисплейном классе), тип операционной системы – Linux, Ubuntu(рис. 4.1).

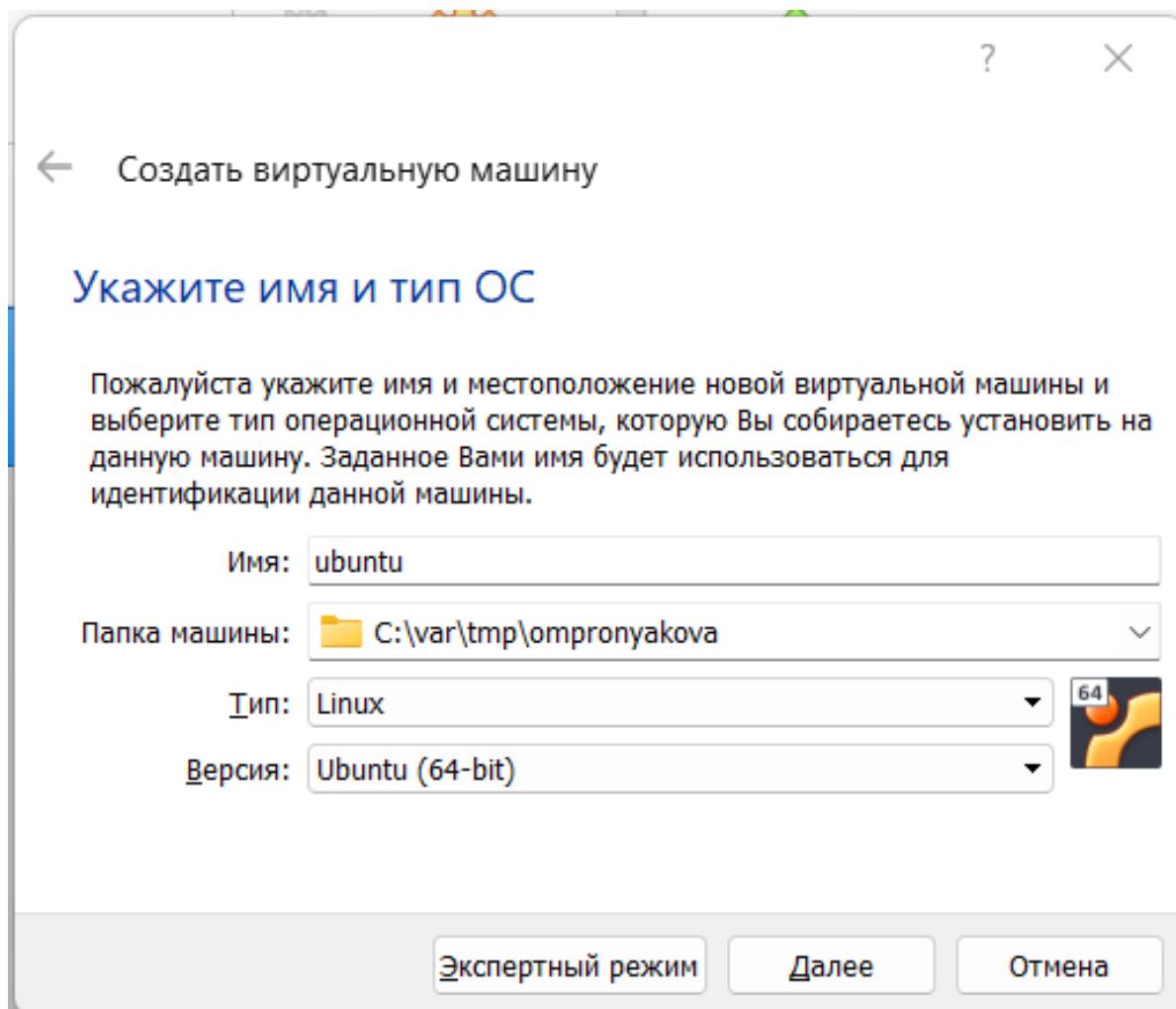


Рис. 4.1: Окно «Имя машины и тип ОС»

Указала размер основной памяти виртуальной машины – 2048 МБ. Задала конфигурацию жёсткого диска – загрузочный, VDI (VirtualBox Disk Image), динамический виртуальный диск. Задала размер диска – 80 ГБ, его расположение – в данном случае /var/tmp/ompronyakova/ubuntu.vdi В настройках виртуальной машины во вкладке Дисплей Экран увеличила доступный объем видеопамяти до 128 МБ(рис. 4.2).

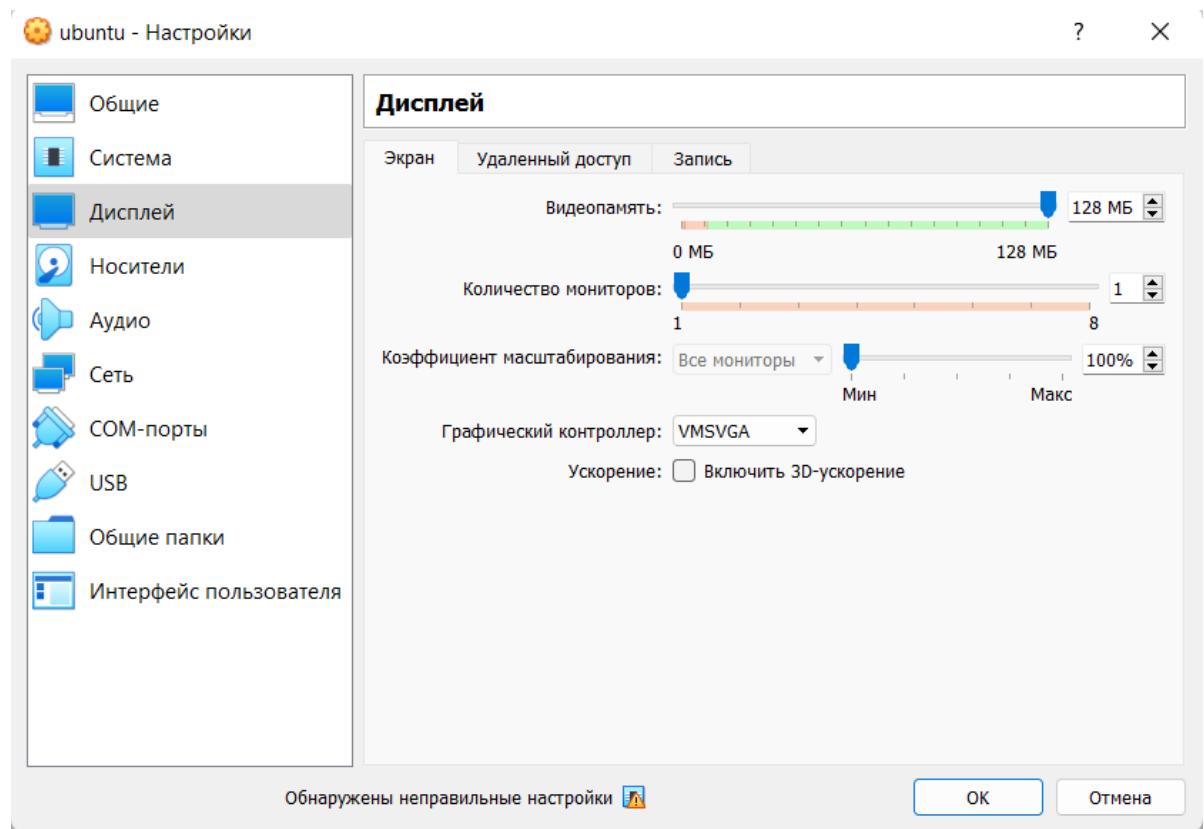


Рис. 4.2: Настройка виртуальной машины

В настройках виртуальной машины во вкладке Носители добавила новый привод оптических дисков(рис. 4.3).

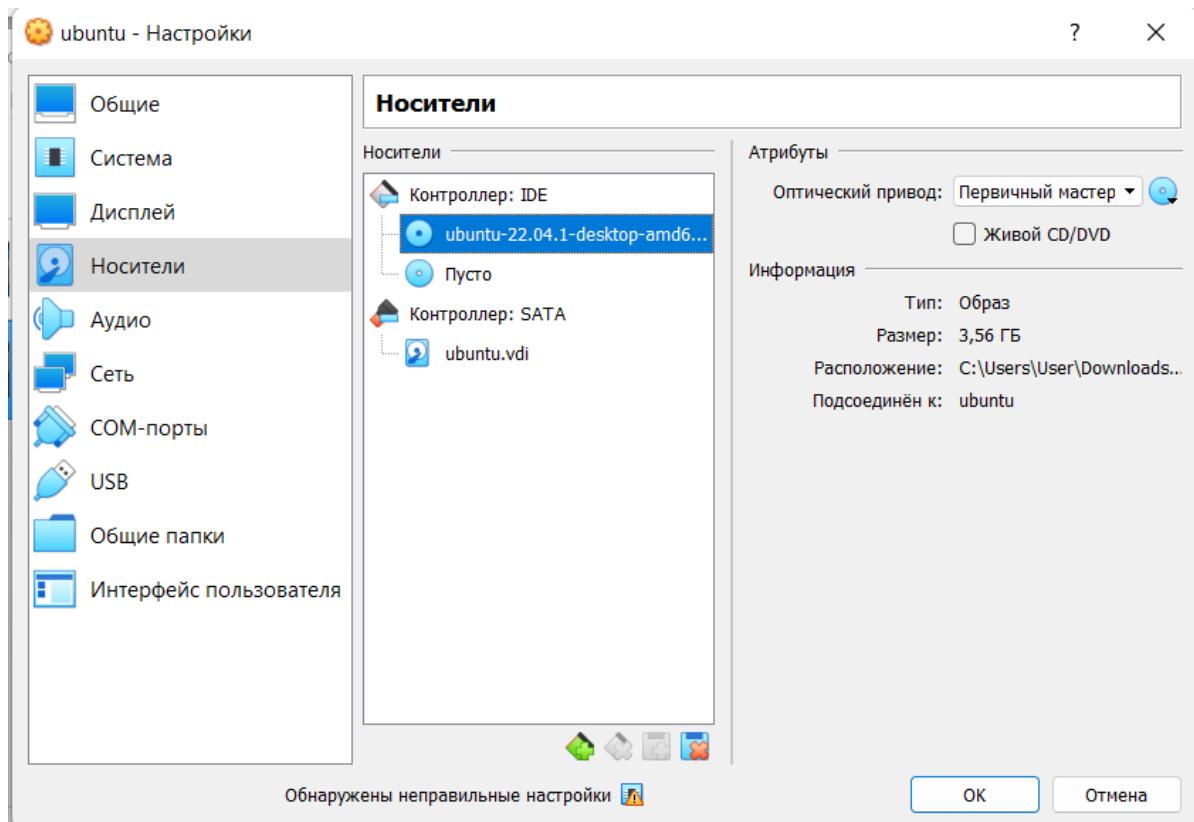


Рис. 4.3: Окно «Носители» виртуальной машины: выбор образа оптического диска

При установке на собственную технику использовала скачанный образ операционной системы Ubuntu:

1. скачанный образ операционной системы Ubuntu

## 4.2 Запуск виртуальной машины и установка системы

Запустила виртуальную машину (Машина Запустить). После загрузки с виртуального оптического диска я увидела окно с двумя вариантами и выбрала установить Ubuntu(рис. 4.4).

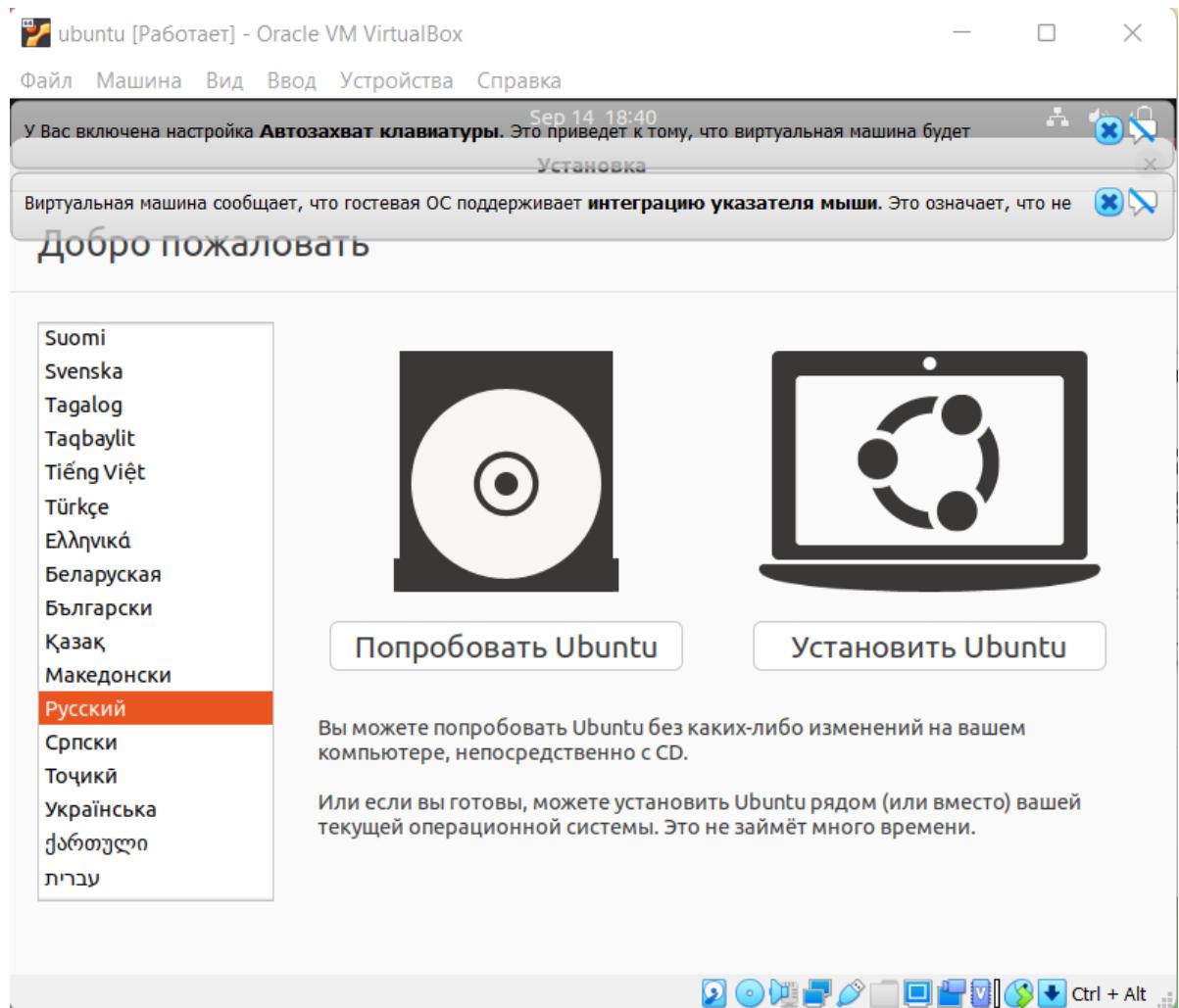


Рис. 4.4: Окно запуска установки образа ОС

Выбрала язык и скорректировала часовой пояс(рис. 4.5). Место установки ОС оставила без изменения. Далее «Тип установки» выбрала «Стереть диск и установить Ubuntu»).

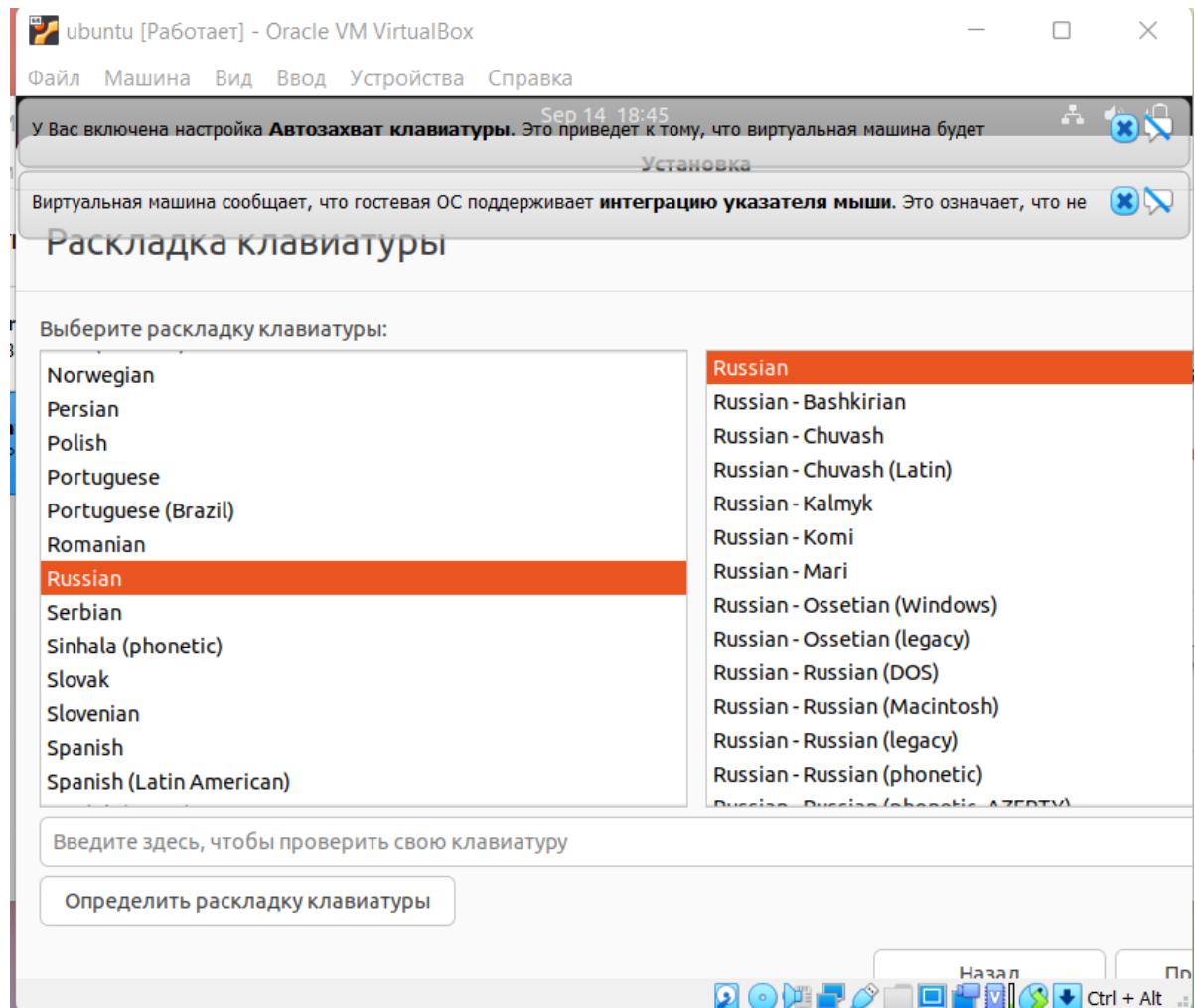


Рис. 4.5: Окно выбора языка

Затем нужно было зарегистрироваться(рис. 4.6).

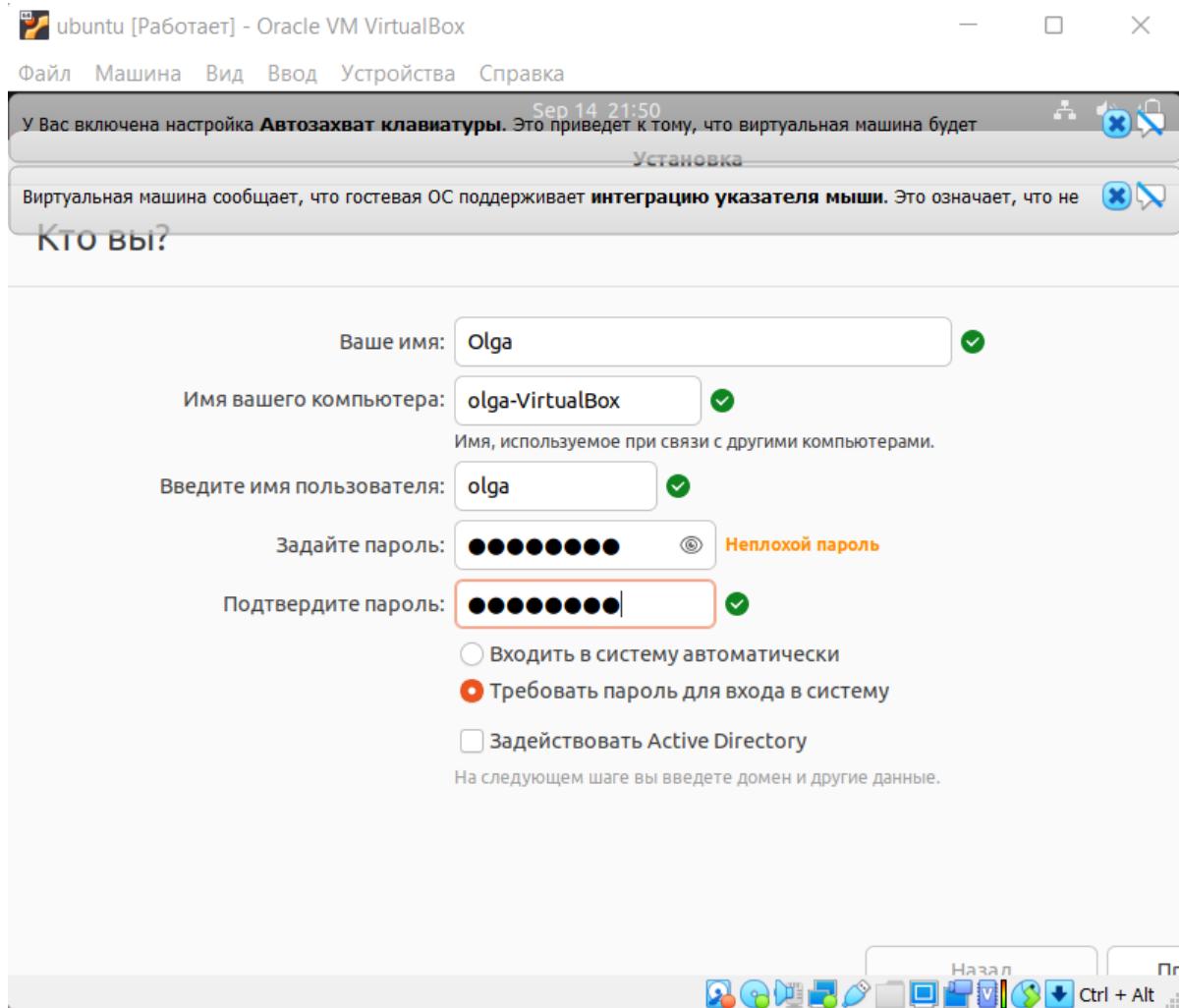


Рис. 4.6: Окно создания пользователя и задания пароля

Затем началась установка Ubuntu. После установки требовалось подтвердить подлинность и ввести пароль, который создавался мною ранее(рис. 4.7).

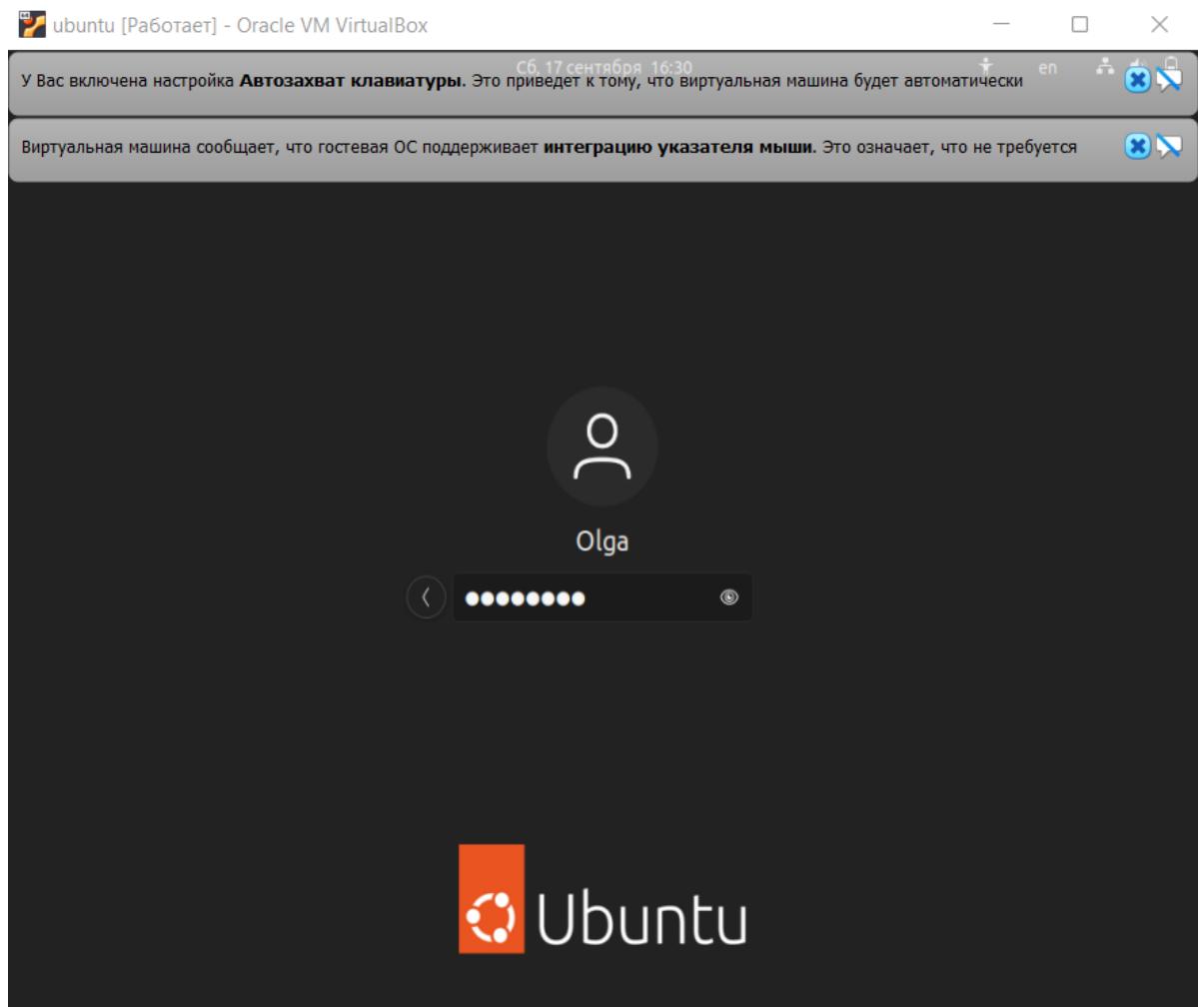


Рис. 4.7: Окно Авторизация

### 4.3 Завершение установки

После окончания установки закрыла окно установщика и выключила систему(рис. 4.8).

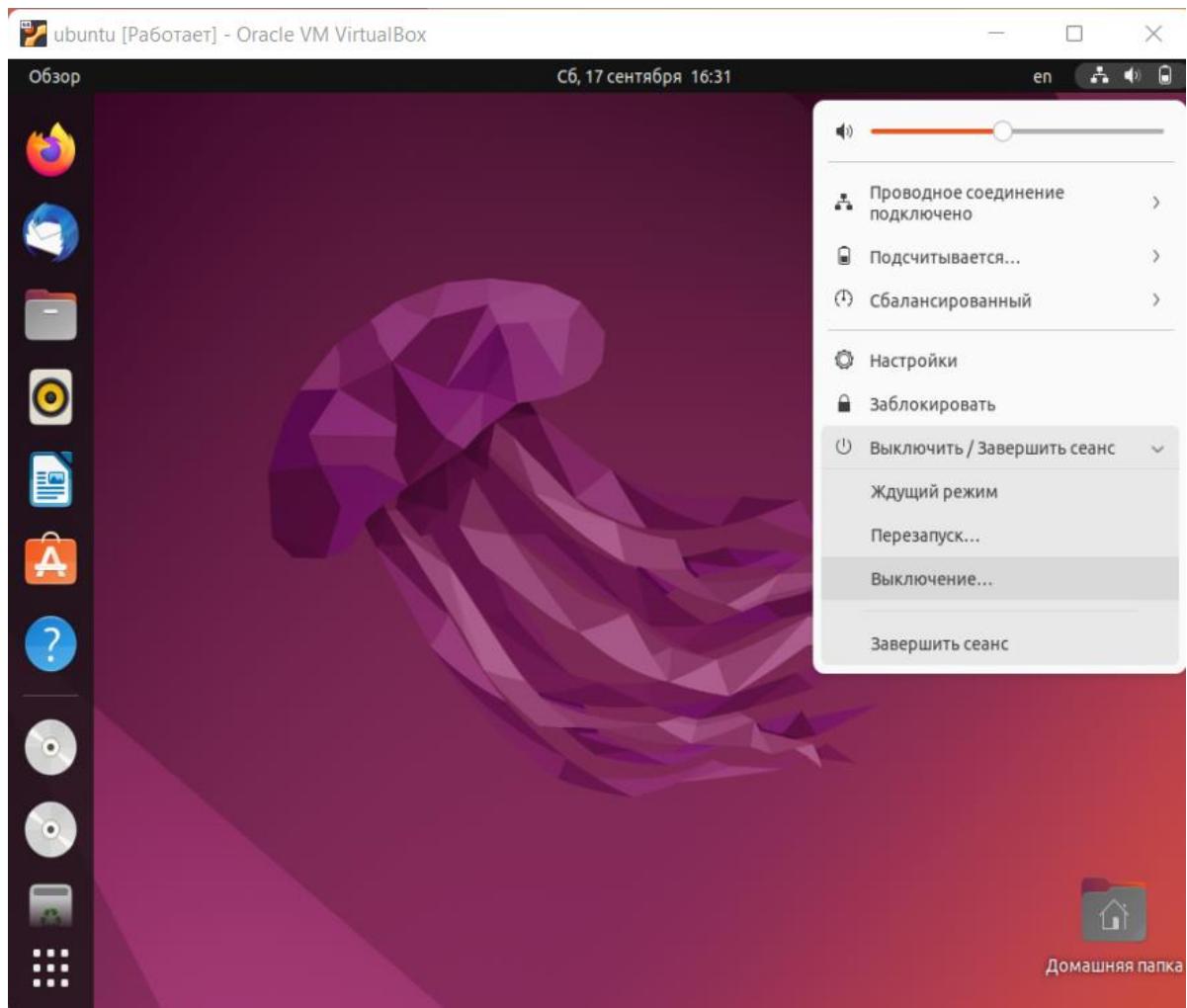


Рис. 4.8: Выключение системы

После того, как виртуальная машина отключилась, следовало изъять образ диска из дисковода. Нажала на значок диска и выбрала пункт изъять. После извлечения в дисководе стало пусто(рис. 4.9).

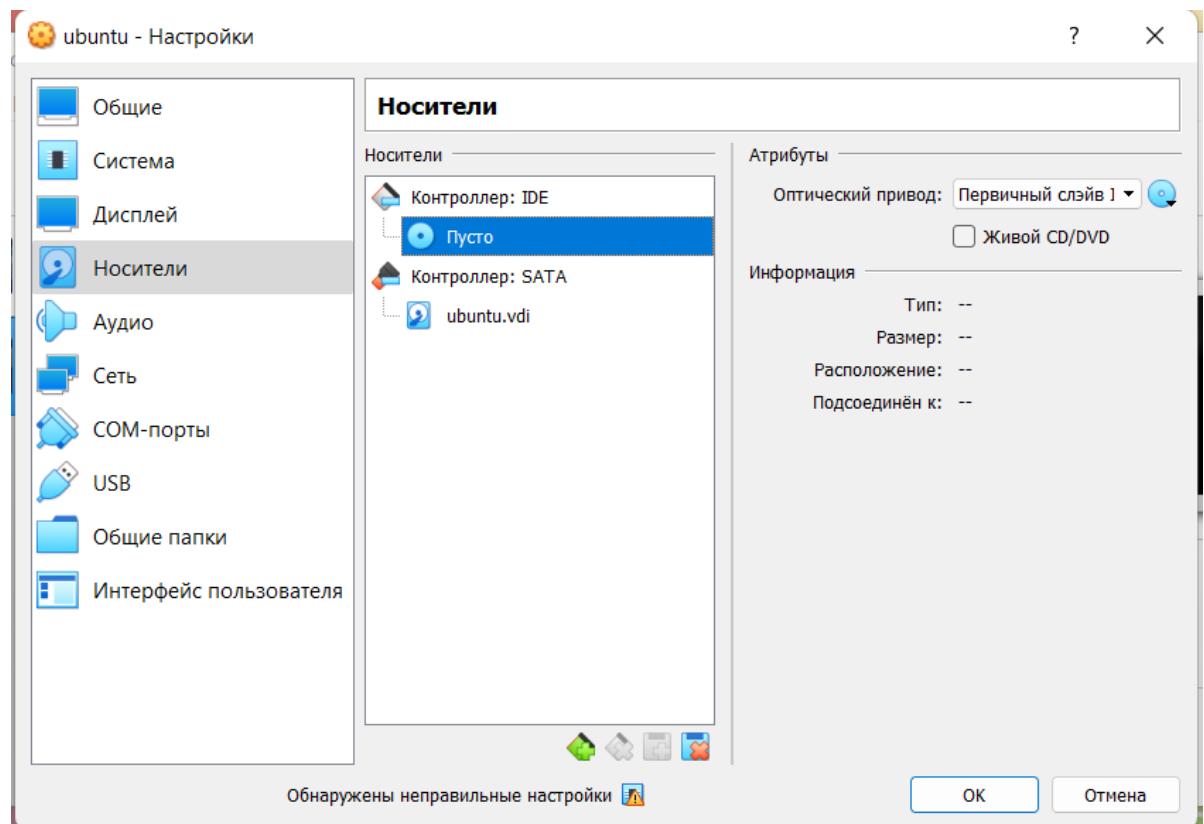


Рис. 4.9: Извлечение образа диска

## 5 Задания для самостоятельной работы

1. Запустила установленную в VirtualBox ОС
2. Нашла в меню приложений и запустите браузер Firefox. Все хорошо открывается(рис. 5.1).

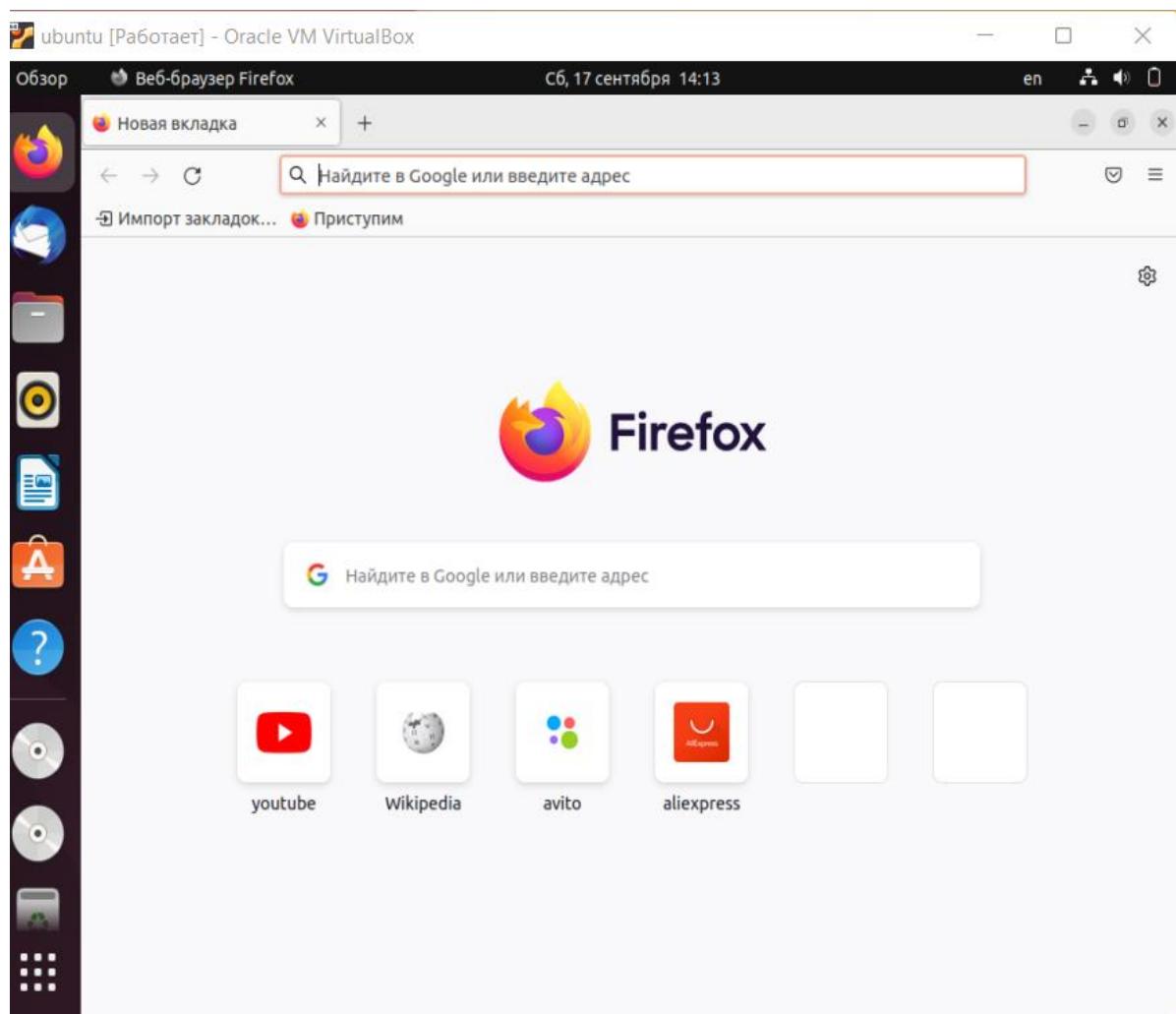


Рис. 5.1: Браузер Firefox

3. Запустила терминал (консоль).
4. Установила основное программное обеспечение необходимое для дальнейшей работы. Программы, которые я вводила:
  - 1) Midninght Commander (mc): :~\$ sudo apt install mc(рис. 5.2). 2)Git: ~\$ sudo apt install git(рис. 5.3). 3)Nasm (Netwide Assembler): ~\$ sudo apt install nasm(рис. 5.4).

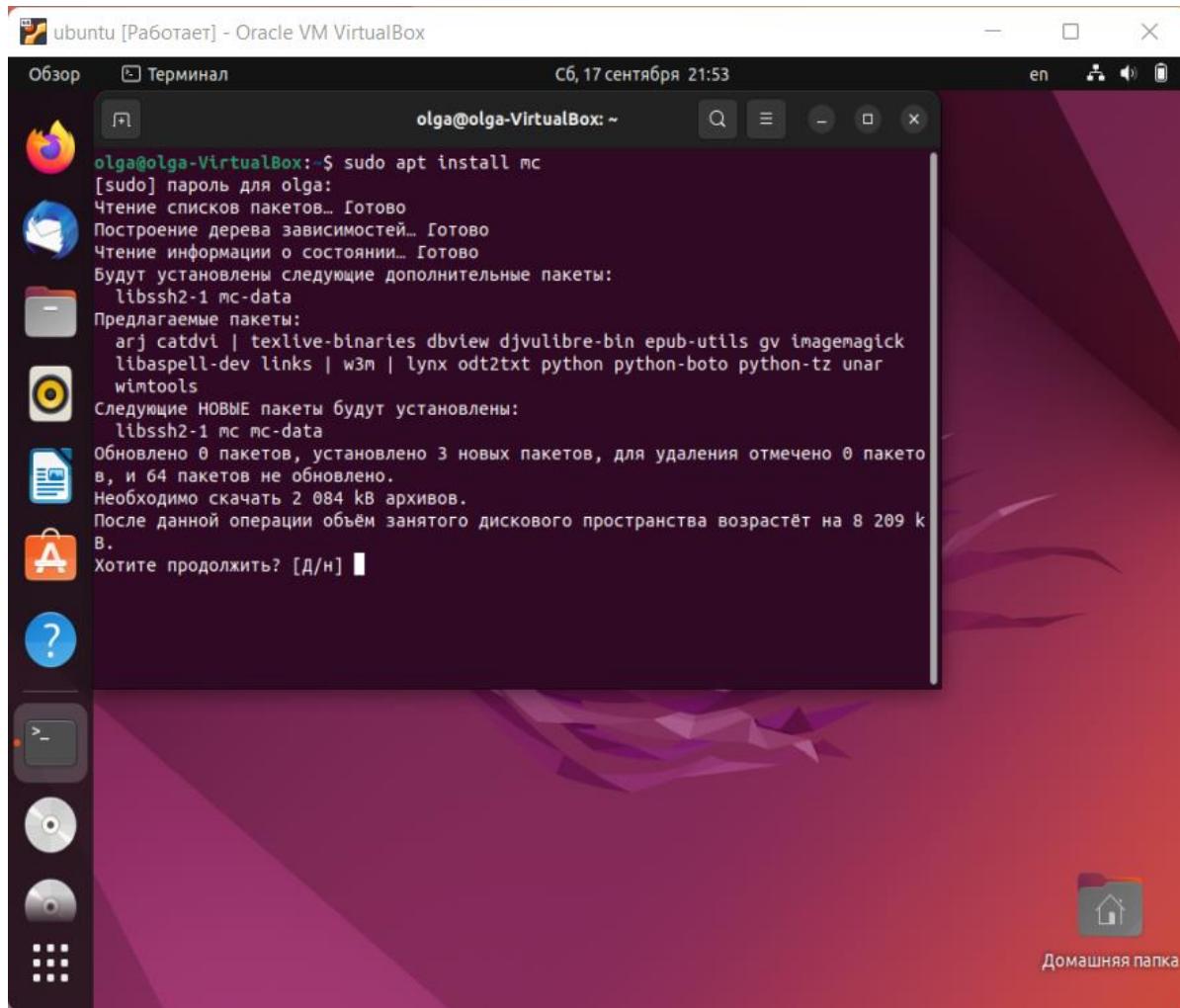


Рис. 5.2: Установила основное программное обеспечение Midninght Commander (mc)

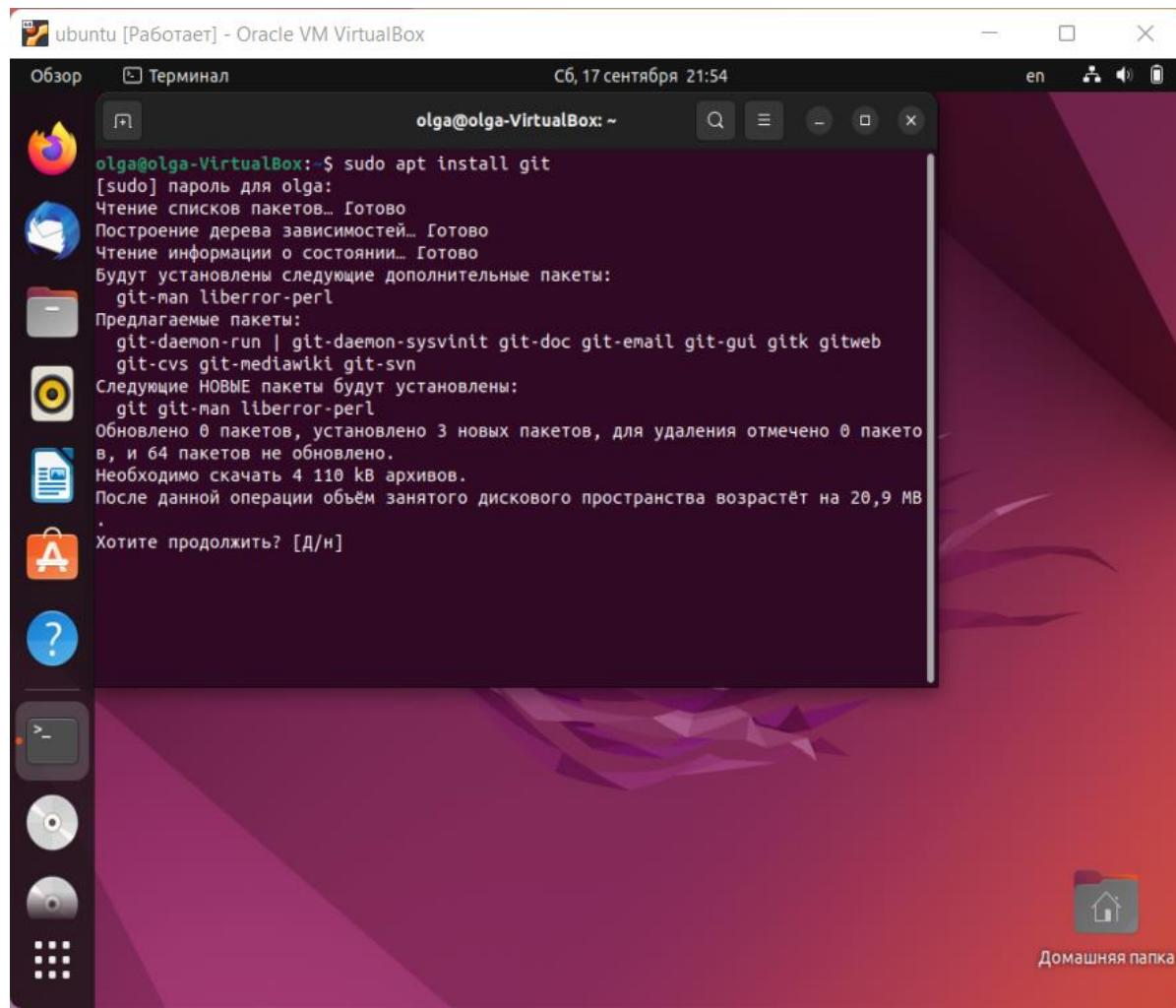


Рис. 5.3: Установила основное программное обеспечение Git

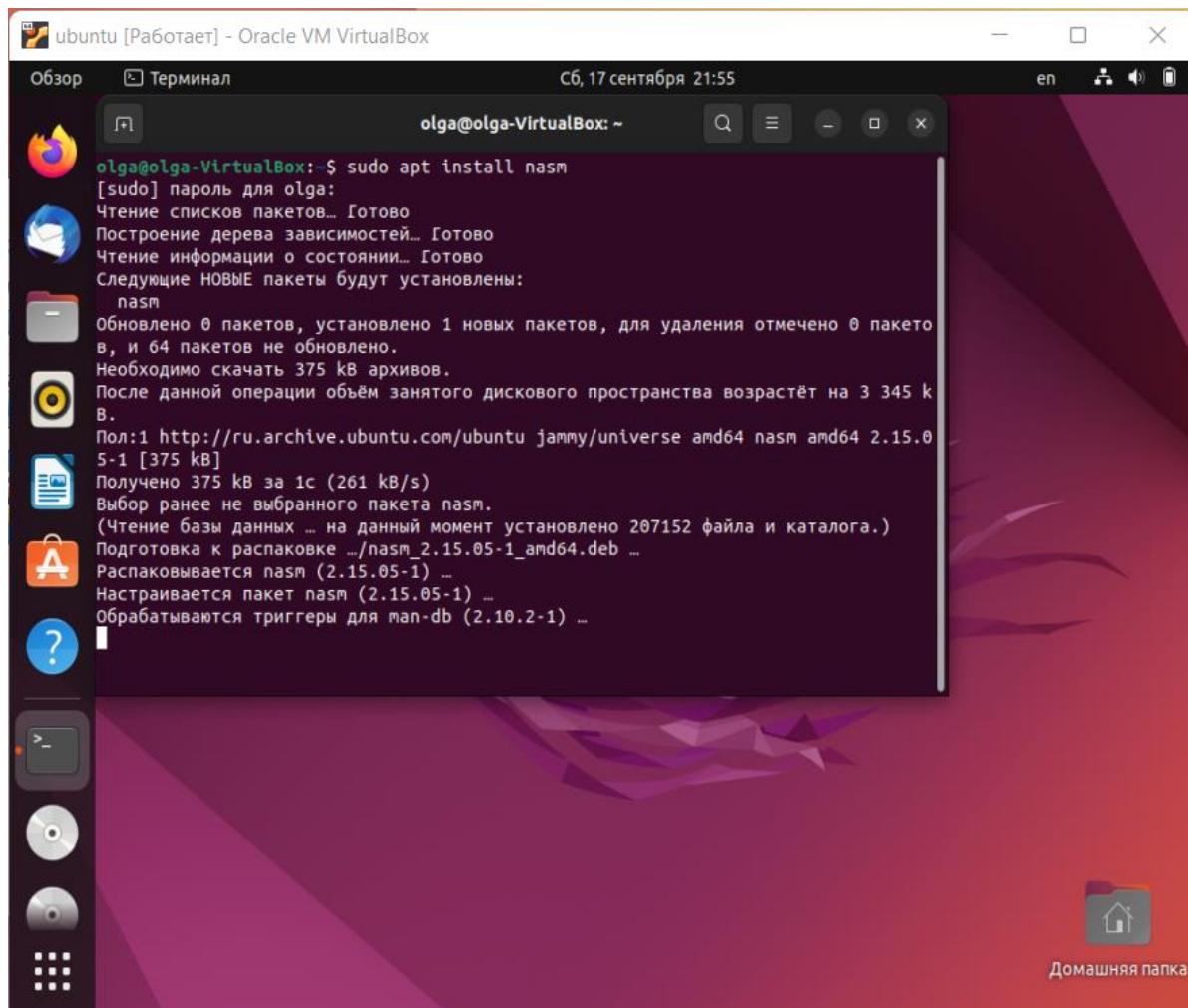
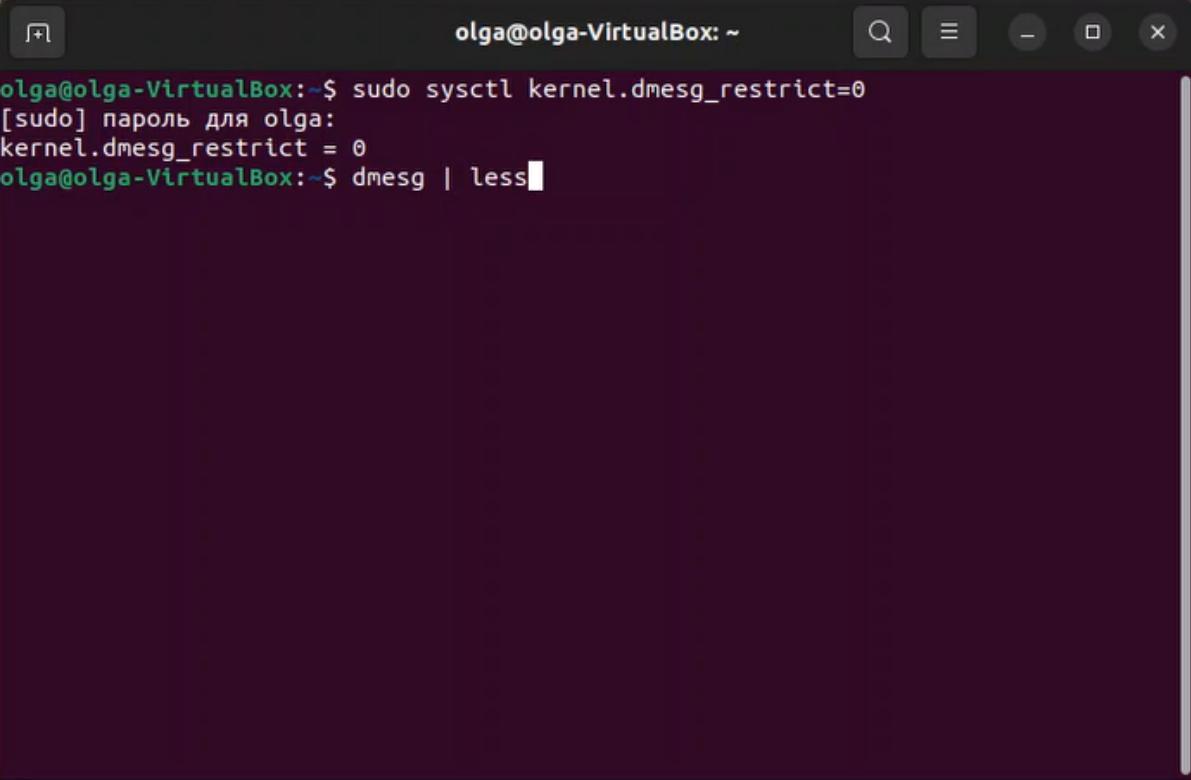


Рис. 5.4: Установила основное программное обеспечение Nasm (Netwide Assembler)

## 6 Выполнение домашней работы

Открыла терминал. В окне терминала проанализировала последовательность загрузки системы, выполнив команду dmesg(рис. 6.1) (рис. 6.2).

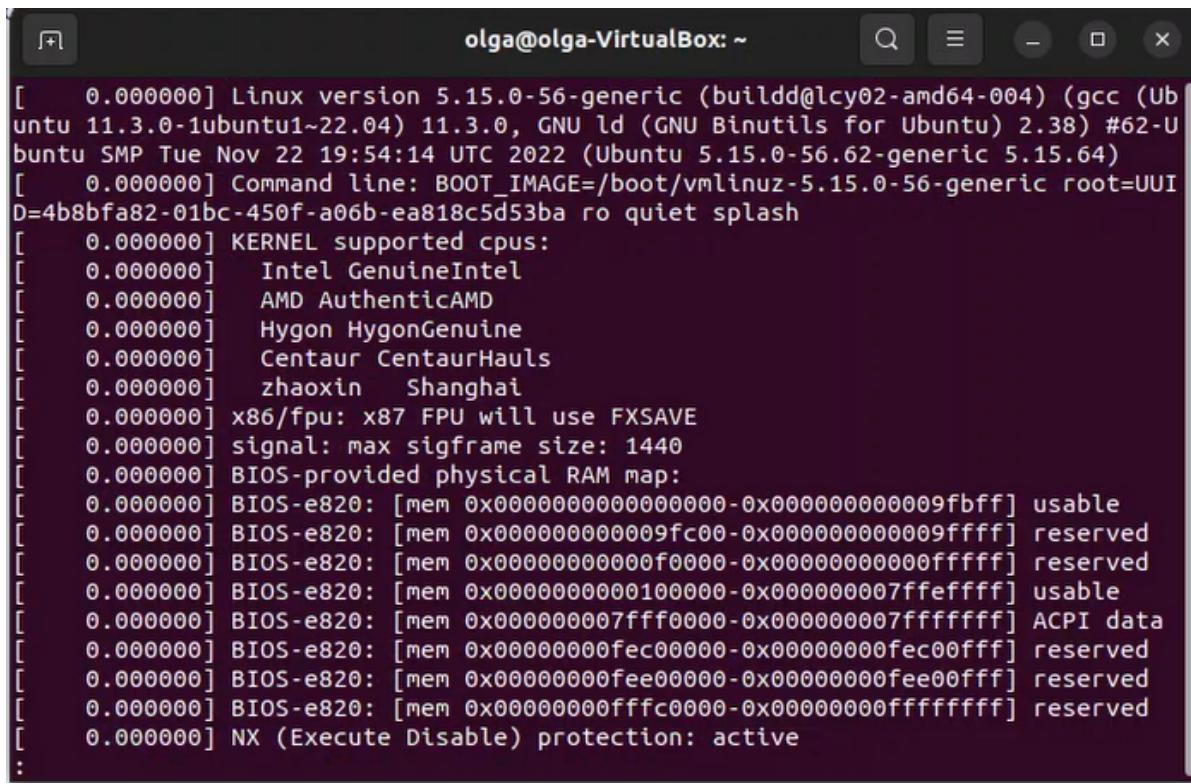


The screenshot shows a terminal window titled "olga@olga-VirtualBox: ~". The window contains the following command-line session:

```
olga@olga-VirtualBox:~$ sudo sysctl kernel.dmesg_restrict=0
[sudo] пароль для olga:
kernel.dmesg_restrict = 0
olga@olga-VirtualBox:~$ dmesg | less
```

The terminal window has a dark background and light-colored text. It includes standard window controls (minimize, maximize, close) at the top right.

Рис. 6.1: Анализ последовательности загрузки системы



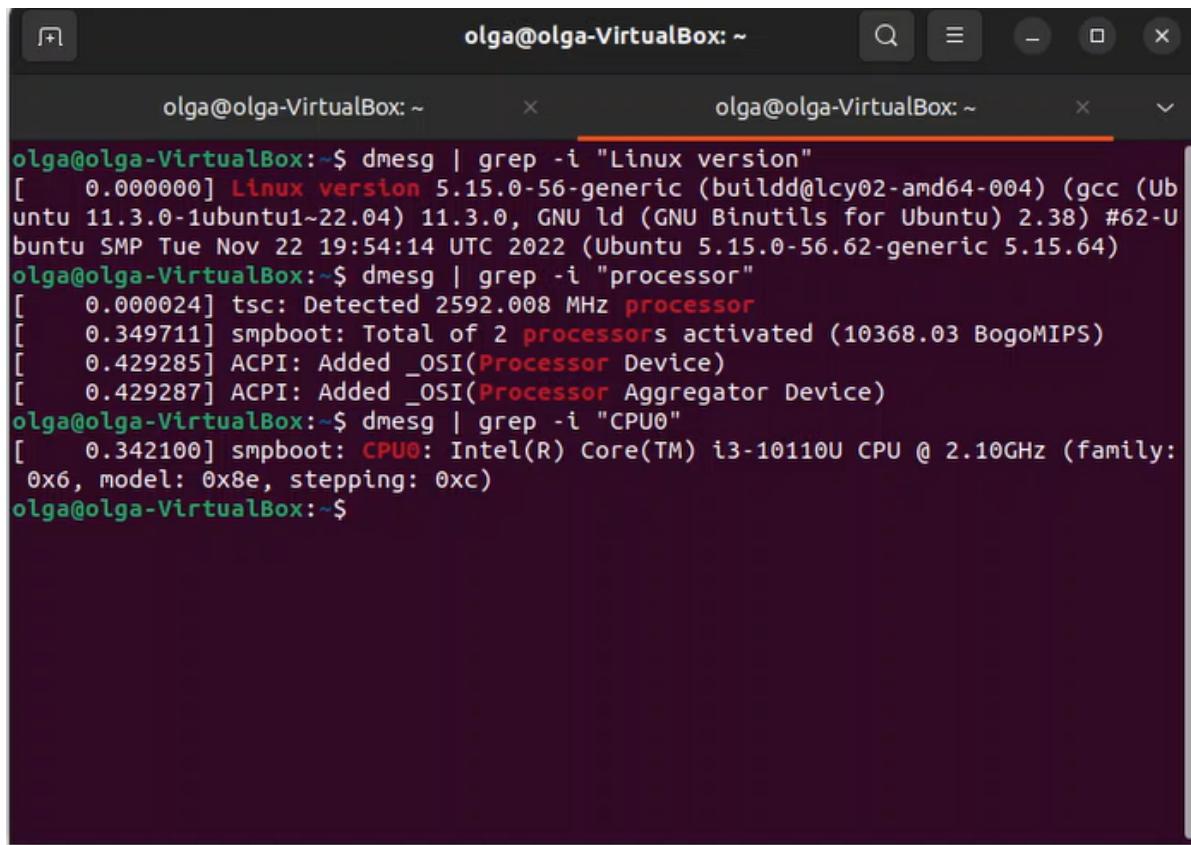
The screenshot shows a terminal window with the title "olga@olga-VirtualBox: ~". The window contains a log of system boot information. The log includes details about the Linux kernel version (5.15.0-56-generic), build date (Tue Nov 22 19:54:14 UTC 2022), command line parameters (BOOT\_IMAGE=/boot/vmlinuz-5.15.0-56-generic root=UUID D=4b8bfa82-01bc-450f-a06b-ea818c5d53ba ro quiet splash), supported cpus (Intel GenuineIntel, AMD AuthenticAMD, Hygon HygonGenuine, Centaur CentaurHauls, zhaixin Shanghai), processor model (x86/fpu: x87 FPU will use FXSAVE), signal handling (max sigframe size: 1440), BIOS-provided physical RAM map, and memory protection (NX (Execute Disable) protection: active). The log ends with a colon ":".

```
[ 0.000000] Linux version 5.15.0-56-generic (buildd@lcy02-amd64-004) (gcc (Ubuntu 11.3.0-1ubuntu1~22.04) 11.3.0, GNU ld (GNU Binutils for Ubuntu) 2.38) #62-Ubuntu SMP Tue Nov 22 19:54:14 UTC 2022 (Ubuntu 5.15.0-56.62-generic 5.15.64)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.15.0-56-generic root=UUID D=4b8bfa82-01bc-450f-a06b-ea818c5d53ba ro quiet splash
[ 0.000000] KERNEL supported cpus:
[ 0.000000]   Intel GenuineIntel
[ 0.000000]   AMD AuthenticAMD
[ 0.000000]   Hygon HygonGenuine
[ 0.000000]   Centaur CentaurHauls
[ 0.000000]   zhaixin Shanghai
[ 0.000000] x86/fpu: x87 FPU will use FXSAVE
[ 0.000000] signal: max sigframe size: 1440
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x000000000000f0000-0x00000000000ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000100000-0x0000000007ffff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000007fff0000-0x0000000007ffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee00fff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000ffffc0000-0x00000000fffffc0000] reserved
[ 0.000000] NX (Execute Disable) protection: active
:
```

Рис. 6.2: Анализ последовательности загрузки системы

Использовала поиск с помощью grep, чтобы получить следующую информацию:

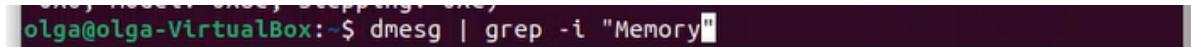
Версия ядра Linux (Linux version); Частота процессора (Detected Mhz processor);  
Модель процессора (CPU0)(рис. 6.3);



The screenshot shows two terminal windows side-by-side. Both windows have the title "olga@olga-VirtualBox: ~". The left window displays the command "olga@olga-VirtualBox:~\$ dmesg | grep -i "Linux version" [ 0.000000] Linux version 5.15.0-56-generic (buildd@lcy02-amd64-004) (gcc (Ubuntu 11.3.0-1ubuntu1~22.04) 11.3.0, GNU ld (GNU Binutils for Ubuntu) 2.38) #62-U buntu SMP Tue Nov 22 19:54:14 UTC 2022 (Ubuntu 5.15.0-56.62-generic 5.15.64)". The right window displays the command "olga@olga-VirtualBox:~\$ dmesg | grep -i "processor" [ 0.000024] tsc: Detected 2592.008 MHz processor [ 0.349711] smpboot: Total of 2 processors activated (10368.03 BogoMIPS) [ 0.429285] ACPI: Added \_OSI(Processor Device) [ 0.429287] ACPI: Added \_OSI(Processor Aggregator Device)". Both windows have a dark background and light-colored text.

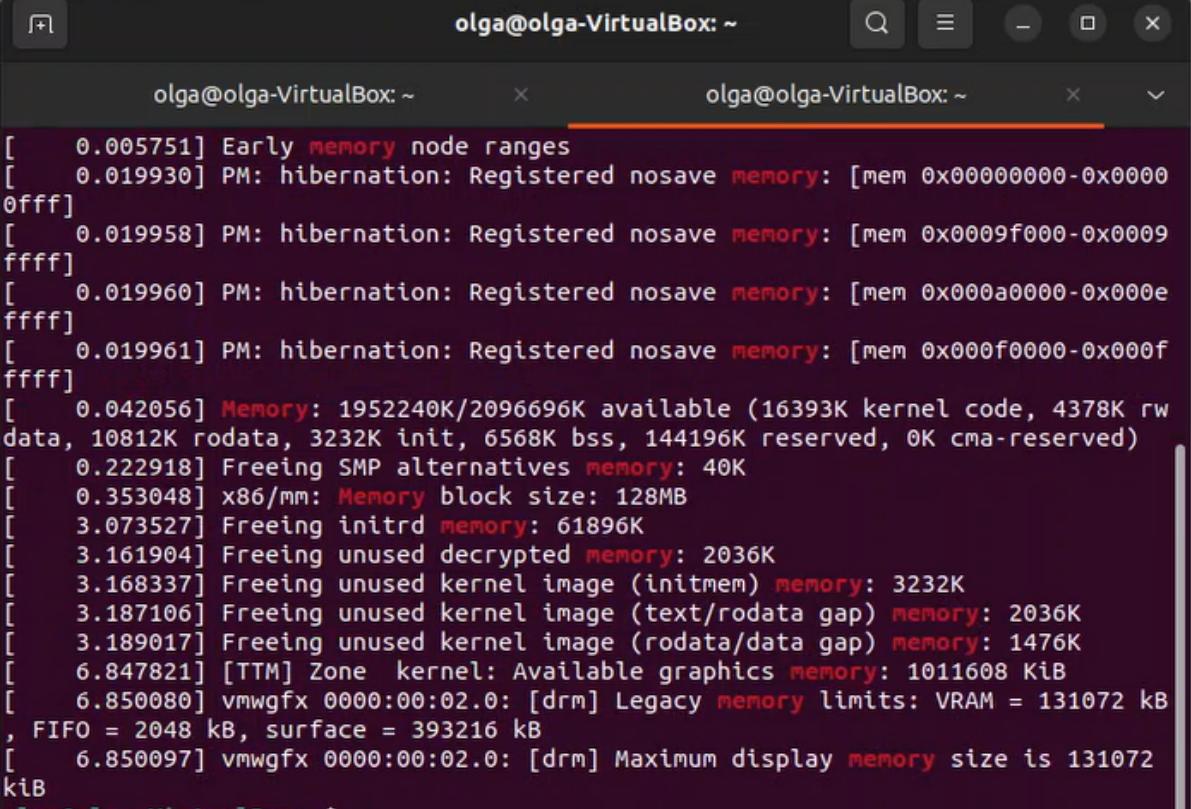
Рис. 6.3: Версия ядра Linux, Частота процессора, Модель процессора

Объём доступной оперативной памяти (Memory available)(рис. 6.4) (рис. 6.5);



The screenshot shows a single terminal window with the title "olga@olga-VirtualBox:~\$ dmesg | grep -i "Memory"". The output of the command is visible at the bottom of the window. The window has a dark background and light-colored text.

Рис. 6.4: Объём доступной оперативной памяти

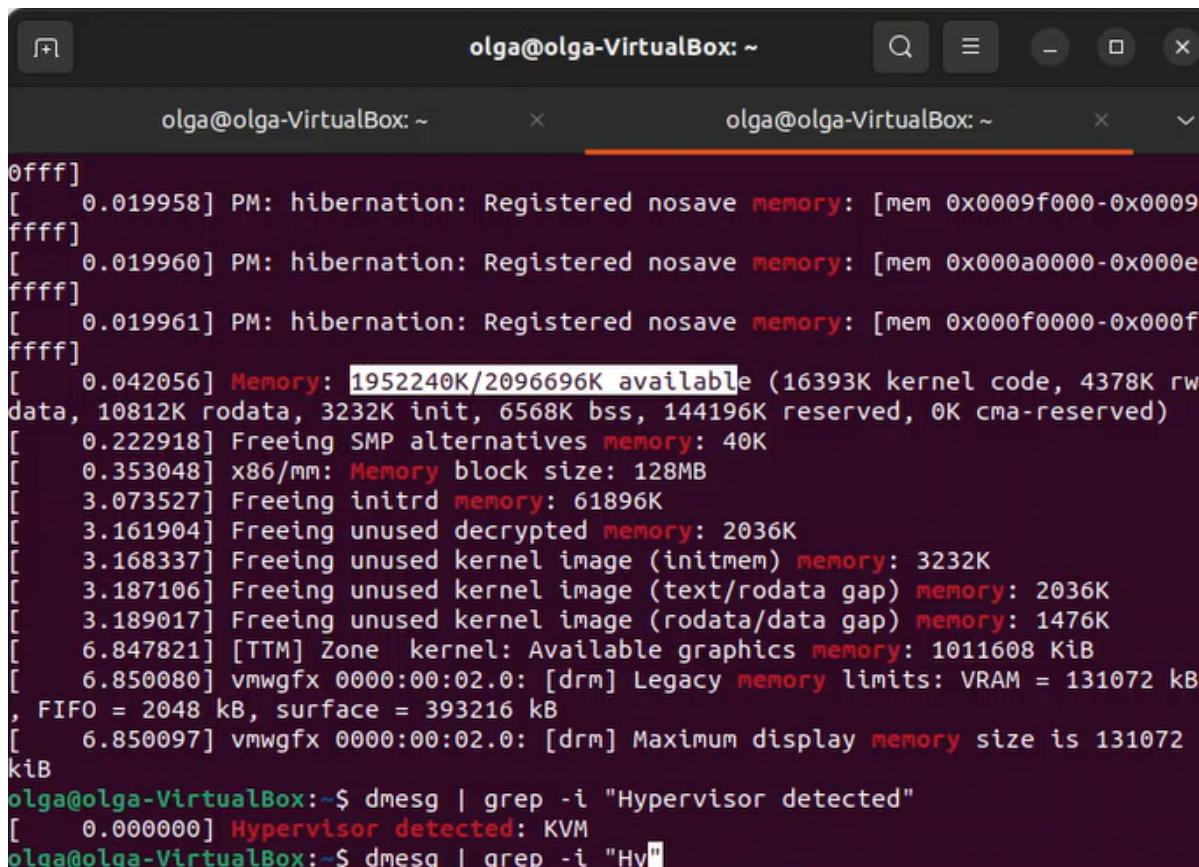


The screenshot shows two terminal windows side-by-side. Both windows have the title "olga@olga-VirtualBox: ~". The left window displays a log of memory initialization and deallocation events, while the right window is mostly blank. The log includes entries such as:

```
[ 0.005751] Early memory node ranges
[ 0.019930] PM: hibernation: Registered nosave memory: [mem 0x00000000-0x0000
0fff]
[ 0.019958] PM: hibernation: Registered nosave memory: [mem 0x0009f000-0x0009
ffff]
[ 0.019960] PM: hibernation: Registered nosave memory: [mem 0x000a0000-0x000e
ffff]
[ 0.019961] PM: hibernation: Registered nosave memory: [mem 0x000f0000-0x000f
ffff]
[ 0.042056] Memory: 1952240K/2096696K available (16393K kernel code, 4378K rw
data, 10812K rodata, 3232K init, 6568K bss, 144196K reserved, 0K cma-reserved)
[ 0.222918] Freeing SMP alternatives memory: 40K
[ 0.353048] x86/mm: Memory block size: 128MB
[ 3.073527] Freeing initrd memory: 61896K
[ 3.161904] Freeing unused decrypted memory: 2036K
[ 3.168337] Freeing unused kernel image (initmem) memory: 3232K
[ 3.187106] Freeing unused kernel image (text/rodata gap) memory: 2036K
[ 3.189017] Freeing unused kernel image (rodata/data gap) memory: 1476K
[ 6.847821] [TTM] Zone kernel: Available graphics memory: 1011608 KiB
[ 6.850080] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 131072 kB
, FIFO = 2048 kB, surface = 393216 kB
[ 6.850097] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 131072
KiB
```

Рис. 6.5: Объём доступной оперативной памяти

Тип обнаруженного гипервизора (Hypervisor detected)(рис. 6.6);

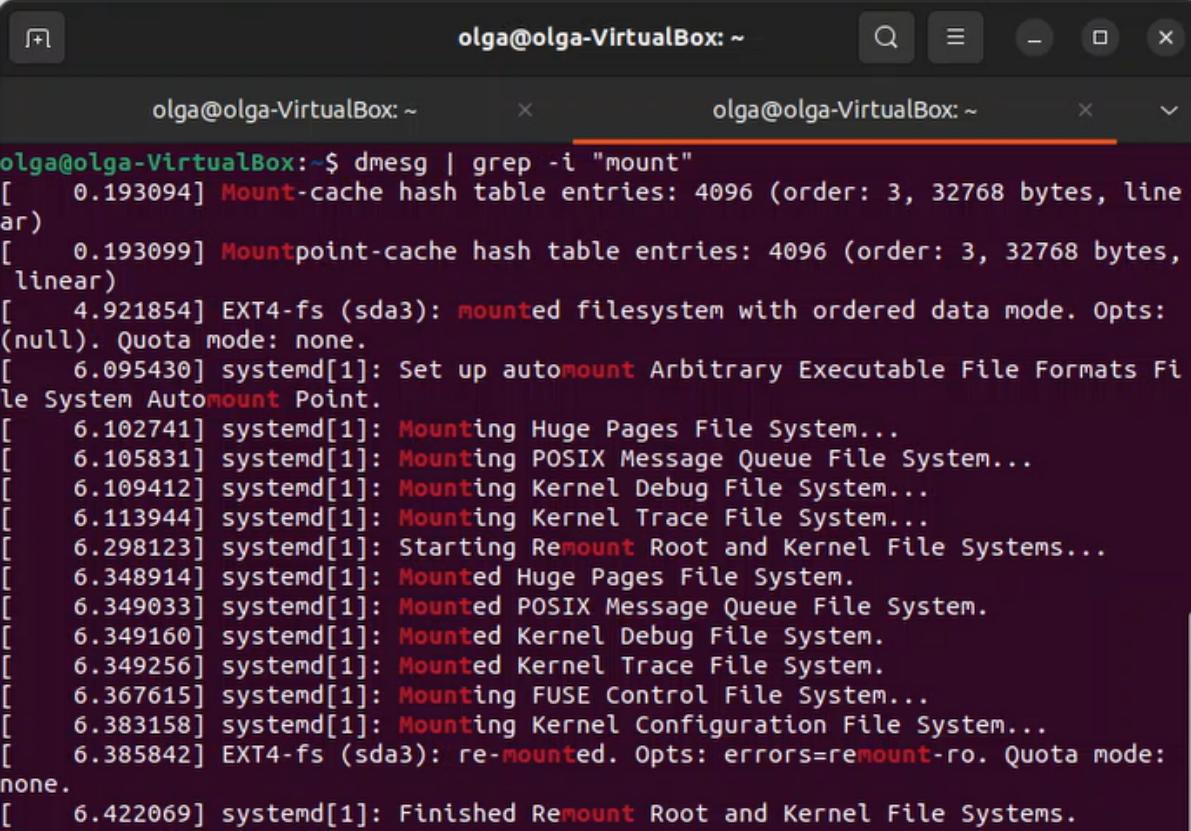


The screenshot shows a terminal window with two tabs, both titled "olga@olga-VirtualBox: ~". The terminal displays the kernel's dmesg log. The log includes messages about hibernation registration, memory management (freeing SMP alternatives, x86/mm, initrd, unused decrypted memory, kernel image gaps), and graphics memory limits. It also shows a message from the Hypervisor detected module indicating that KVM is present. The terminal prompt at the bottom is "olga@olga-VirtualBox:~\$".

```
0fff]
[    0.019958] PM: hibernation: Registered nosave memory: [mem 0x00009f000-0x00009ffff]
[    0.019960] PM: hibernation: Registered nosave memory: [mem 0x000a00000-0x000e0fff]
[    0.019961] PM: hibernation: Registered nosave memory: [mem 0x000f00000-0x000ff0fff]
[    0.042056] Memory: 1952240K/2096696K available (16393K kernel code, 4378K rwdata, 10812K rodata, 3232K init, 6568K bss, 144196K reserved, 0K cma-reserved)
[    0.222918] Freeing SMP alternatives memory: 40K
[    0.353048] x86/mm: Memory block size: 128MB
[    3.073527] Freeing initrd memory: 61896K
[    3.161904] Freeing unused decrypted memory: 2036K
[    3.168337] Freeing unused kernel image (initmem) memory: 3232K
[    3.187106] Freeing unused kernel image (text/rodata gap) memory: 2036K
[    3.189017] Freeing unused kernel image (rodata/data gap) memory: 1476K
[    6.847821] [TTM] Zone kernel: Available graphics memory: 1011608 KiB
[    6.850080] vmwgfx 0000:00:02.0: [drm] Legacy memory limits: VRAM = 131072 kB
, FIFO = 2048 kB, surface = 393216 kB
[    6.850097] vmwgfx 0000:00:02.0: [drm] Maximum display memory size is 131072
KiB
olga@olga-VirtualBox:~$ dmesg | grep -i "Hypervisor detected"
[    0.000000] Hypervisor detected: KVM
olga@olga-VirtualBox:~$ dmesg | grep -i "Hy"
```

Рис. 6.6: Тип обнаруженного гипервизора

Тип файловой системы корневого раздела(рис. 6.7);

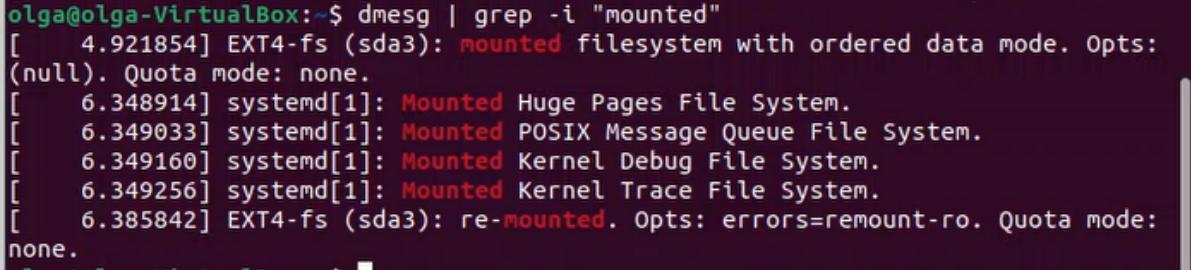


The screenshot shows a terminal window with two tabs, both titled "olga@olga-VirtualBox: ~". The terminal displays the dmesg command output, specifically filtered by grep -i "mount". The log entries show the kernel mounting various file systems, including the root filesystem (EXT4-fs) and other kernel-related file systems like Huge Pages, POSIX Message Queue, Kernel Debug, and Kernel Trace. The output is timestamped and shows the sequence of mounts.

```
olga@olga-VirtualBox:~$ dmesg | grep -i "mount"
[ 0.193094] Mount-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 0.193099] Mountpoint-cache hash table entries: 4096 (order: 3, 32768 bytes, linear)
[ 4.921854] EXT4-fs (sda3): mounted filesystem with ordered data mode. Opts: (null). Quota mode: none.
[ 6.095430] systemd[1]: Set up automount Arbitrary Executable File Formats File System Automount Point.
[ 6.102741] systemd[1]: Mounting Huge Pages File System...
[ 6.105831] systemd[1]: Mounting POSIX Message Queue File System...
[ 6.109412] systemd[1]: Mounting Kernel Debug File System...
[ 6.113944] systemd[1]: Mounting Kernel Trace File System...
[ 6.298123] systemd[1]: Starting Remount Root and Kernel File Systems...
[ 6.348914] systemd[1]: Mounted Huge Pages File System.
[ 6.349033] systemd[1]: Mounted POSIX Message Queue File System.
[ 6.349160] systemd[1]: Mounted Kernel Debug File System.
[ 6.349256] systemd[1]: Mounted Kernel Trace File System.
[ 6.367615] systemd[1]: Mounting FUSE Control File System...
[ 6.383158] systemd[1]: Mounting Kernel Configuration File System...
[ 6.385842] EXT4-fs (sda3): re-mounted. Opts: errors=remount-ro. Quota mode: none.
[ 6.422069] systemd[1]: Finished Remount Root and Kernel File Systems.
olga@olga-VirtualBox:~$
```

Рис. 6.7: Тип файловой системы корневого раздела

Последовательность монтирования файловых систем(рис. 6.8);



The screenshot shows a terminal window with one tab titled "olga@olga-VirtualBox: ~". The terminal displays the dmesg command output, specifically filtered by grep -i "mounted". The log entries show the kernel mounting various file systems, including the root filesystem (EXT4-fs) and other kernel-related file systems like Huge Pages, POSIX Message Queue, Kernel Debug, and Kernel Trace. The output is timestamped and shows the sequence of mounts.

```
olga@olga-VirtualBox:~$ dmesg | grep -i "mounted"
[ 4.921854] EXT4-fs (sda3): mounted filesystem with ordered data mode. Opts: (null). Quota mode: none.
[ 6.348914] systemd[1]: Mounted Huge Pages File System.
[ 6.349033] systemd[1]: Mounted POSIX Message Queue File System.
[ 6.349160] systemd[1]: Mounted Kernel Debug File System.
[ 6.349256] systemd[1]: Mounted Kernel Trace File System.
[ 6.385842] EXT4-fs (sda3): re-mounted. Opts: errors=remount-ro. Quota mode: none.
olga@olga-VirtualBox:~$
```

Рис. 6.8: Последовательность монтирования файловых систем

## 7 Ответы на контрольные вопросы

1. Какую информацию содержит учётная запись пользователя? Имя пользователя, зашифрованный пароль пользователя, идентификационный номер пользователя, идентификационный номер группы пользователя, домашний каталог пользователя, командный интерпретатор пользователя.
2. Укажите команды терминала и приведите примеры: – для получения справки по команде; – для перемещения по файловой системе; – для просмотра содержимого каталога; – для определения объёма каталога; – для создания / удаления каталогов / файлов; – для задания определённых прав на файл / каталог; – для просмотра истории команд.
  - a) для получения справки по команде: man
  - b) для перемещения по файловой системе: cd
  - c) для просмотра содержимого каталога: ls
  - d) для определения объёма каталога: du
  - e) для создания каталогов: mkdir
  - f) для создания файлов: touch
  - g) для удаления каталогов: rm
  - h) для удаления файлов: rm –r
  - i) для задания определённых прав на файл / каталог: chmod + x
  - j) для просмотра истории команд: history
3. Что такое файловая система? Приведите примеры с краткой характеристикой. Файловая система — это часть операционной системы, назначение

которой состоит в том, чтобы обеспечить пользователю удобный интерфейс при работе с данными, хранящимися на диске, и обеспечить совместное использование файлов несколькими пользователями и процессами. Примеры файловых систем:

- Ext2, Ext3, Ext4 или Extended Filesystem – стандартная файловая система для Linux.
- JFS или Journaled File System была разработана в IBM для AIX UNIX и использовалась в качестве альтернативы для файловых систем ext. Она используется там, где необходима высокая стабильность и минимальное потребление ресурсов.
- ReiserFS – была разработана намного позже, но в качестве альтернативы ext3 с улучшенной производительностью и расширенными возможностями.
- XFS – это высокопроизводительная файловая система. Преимущества: высокая скорость работы с большими файлами, отложенное выделение места, увеличение разделов на лету и незначительный размер служебной информации.

4. Как посмотреть, какие файловые системы подмонтированы в ОС? С помощью команды `mount`.
5. Как удалить зависший процесс? С помощью команды `kill`.

## **8 Выводы**

В ходе данной лабораторной работы я приобрела практические навыки установки операционной системы на виртуальную машину, настройки минимально необходимых для дальнейшей работы сервисов.

# **Список литературы**

1. Архитектура ЭВМ
2. Youtube(Как установить Ubuntu на VirtualBox)