

*Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования*

АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет математики и информатики

Т. А. Галаган, Е.В. Дегтярев

**Методические указания
к лабораторным работам по дисциплине
«Операционные системы»**

Благовещенск

2021

Галаган Т. А. Методические указания к лабораторным работам по дисциплине «Операционные системы» для направлений подготовки 09.03.01 Информатика и вычислительная техника, 09.03.02 Информационные системы и технологии, 09.03.04 Программная инженерия, 01.03.02 Прикладная математика и информатика /Т. А. Галаган, Е. В. Дегтярев – Благовещенск: Амурский гос. ун-т, 2021. – 74 с.

Учебно-методическое пособие содержит описание пяти лабораторных работ, каждая из которых включает краткие теоретические сведения, технологию выполнения работ, индивидуальные варианты заданий для самостоятельного выполнения, контрольные вопросы. Выполнение лабораторных работ, изложенных в пособии, позволяет приобретать умения и практические навыки инсталляции современных операционных системах и работы с ними.

Рецензент:

Чалкина Н.А., канд. пед. наук, доцент кафедры общей математики и информатики ФГБОУ ВО «Амурский государственный университет».

© Амурский государственный университет, 2021

© Кафедра информационных и управляющих систем, 2021

© Галаган Т.А., Дегтярев Е.В.. авторы

ВВЕДЕНИЕ

Цель освоения дисциплины «Операционные системы» – изучение принципов построения, назначения, теоретических основ функционирования и практического использования операционных систем как эффективного средства управления процессами обработки данных в современных программно-аппаратных комплексах автоматизированных и информационных систем.

Задачи дисциплины:

- изучение принципов управления задачами, процессами и ресурсами в операционных системах, алгоритмов распределения основной памяти, подходов организации управления вводом-выводом, принципов работы файловых систем;

- приобретение умений и практических навыков инсталляции современных операционных системах семейства Windows и Linux, операционных систем мобильных устройств Android, проверки их работоспособности, параметрической настройки.

Общепрофессиональными компетенциями обучающегося, формируемыми в результате освоения дисциплины «Операционные системы», являются:

ОПК-5 Способен инсталлировать программное и аппаратное обеспечение,

ОПК-7 Способен участвовать в настройке и наладке программно-аппаратных комплексов.

Данное учебно-методическое пособие предназначено для проведения лабораторных работ по указанной дисциплине. Материал пособия содержит 5 лабораторных работ, каждая из которых включает пошаговое изложение

технологии выполнения задания, индивидуальные варианты заданий, контрольные вопросы.

Лабораторная работа №1

Тема: Создание, установка и запуск виртуальной машины в Oracle VirtualBox.

Цель: Получение практических навыков работы с программным продуктом OracleVirtualBox виртуализации для различных операционных систем.

Задание

1. Создайте виртуальную машину.
2. Выполните настройку гостевой операционной системы.
3. Настройте доступ к сети Интернет.
4. Выполните запуск созданной виртуальной машины.

Технология выполнения работы

VirtualBox позволяет запускать различные виды операционных систем (включая все программное обеспечение работающее в них) непосредственно в рабочей операционной системе, в специальной среде, называемой "виртуальная машина". При этом физический компьютер, на котором проходит работа, обычно называют "главным компьютером" или "хостом"(host), в то время как виртуальную машину часто называют "гостем" или "гостевой ОС"(guest).

Методики и возможности, которые обеспечивает VirtualBox, возможно использовать для следующих сценариев:

Поддержка ОС. В VirtualBox возможно запускать программы, созданные для другой операционной системы (например, программы для Windows на Linux системах) без необходимости загрузки этой ОС. Можно также установить устаревшие ОС, такие как XP или DOS.

Консолидация инфраструктуры (Infrastructure consolidation). Виртуализация может значительно уменьшить расходы на аппаратные средства и электричество. Мощность систем, обеспеченная современными аппаратными

средствами, редко задействована полностью, типичный сервер обычно использует половину своей теоретической мощности. Так, вместо использования нескольких физических компьютеров, которые только частично загружены, вы можете запускать несколько виртуальных машин на мощных главных компьютерах и распределить нагрузку между ними. VirtualBox может работать как простой сервер VirtualBox Remote Desktop Protocol (VRDP – протокол удаленного рабочего стола) с поддержкой USB. Это позволяет сопровождать всю программную инфраструктуру предприятия только на нескольких серверах RDP (терминальный сервер), а фактически клиентским системам достаточно только быть клиентом VRDP(тонкий клиент).

Тестирование и восстановление в аварийных ситуациях. После установки и настройки, виртуальную машину и ее виртуальный жесткий диск можно считать "контейнером", который может быть "заморожен", "разбуден", скопирован и перенесен на другие компьютеры. Вдобавок к этому, используя механизм VirtualBox, называемый "снимки системы" (snapshots), можно сохранить состояние виртуальной машины и "откатываться" назад к этому состоянию, в случае необходимости. Можно свободно экспериментировать с вычислительной средой. Если что-то пойдет не так, как надо (например, после неправильной установки программного обеспечения или заражения гостевой ОС с вирусом), можно легко переключиться назад на предыдущий снимок системы, не выполняя частых резервных копий и их восстановлений.

Virtual machine (VM) – специальная среда, которую VirtualBox создает для гостевой операционной системы. Другими словами, запуск гостевой ОС осуществляется в VM. Обычно, VM отображаются в виде окна на рабочем столе компьютера, возможно их отображение в полноэкранном режиме или в удаленной среде используя технологию удаленного рабочего стола (RDP).

Термин "виртуальная машина" может использоваться и более абстрактно, как набор параметров, которые определяют настройки VirtualBox. Эти параметры отображаются и настраиваются в графическом интерфейсе

пользователя VirtualBox и интерфейсе командной строки VBoxManage;

Запуск VirtualBox зависит от ОС главного компьютера. На Windows хосте, в стандартном меню "Программы", требуется кликнуть на иконке в группе "VirtualBox".

Создание виртуальной машины

Клик на кнопке "New/Создать" (рис. 1) в главном окне инициирует процесс создания новой виртуальной машины.

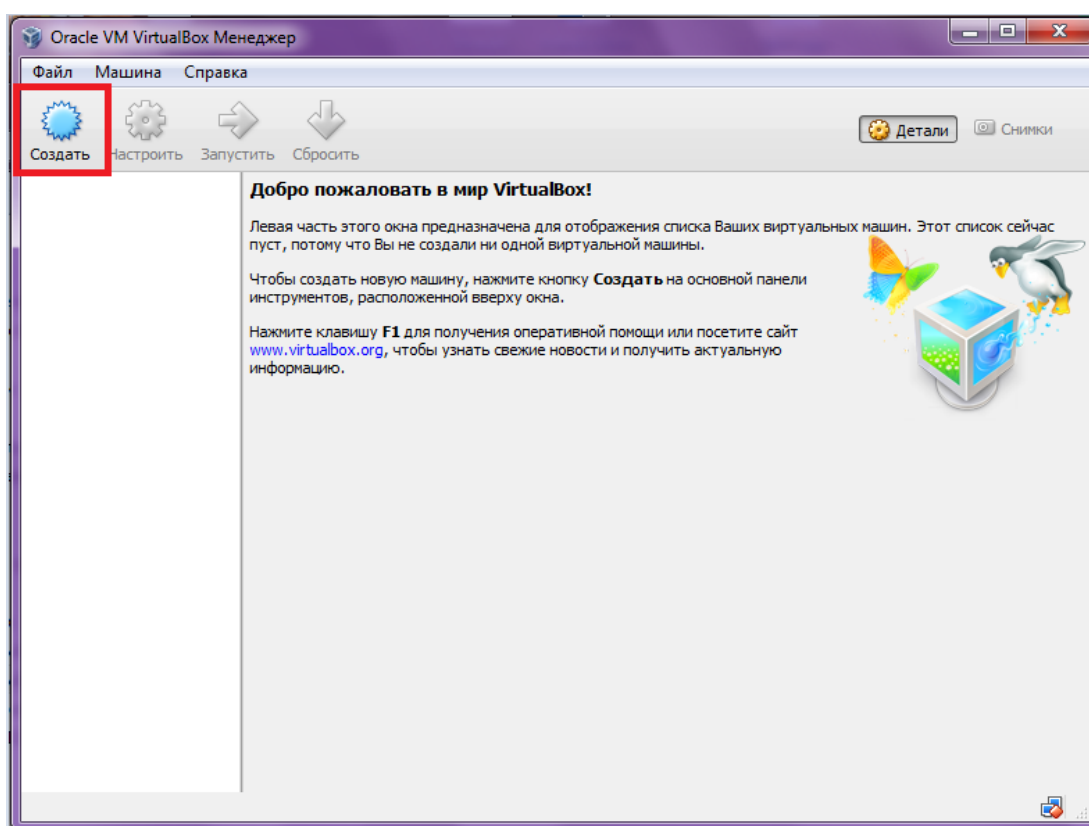


Рисунок 1 – Экранная форма «Главное окно»

На следующих страницах (рис. 2), мастер создания запросит минимум информации, необходимой для создания виртуальной машины, в частности:

- имя виртуальной машины – имя, под которым виртуальная машина будет отображена в главном окне;
- тип – ОС, которую требуется установить;
- версия – версия операционной системы, которая будет установ-

ливаться на создаваемую машину.

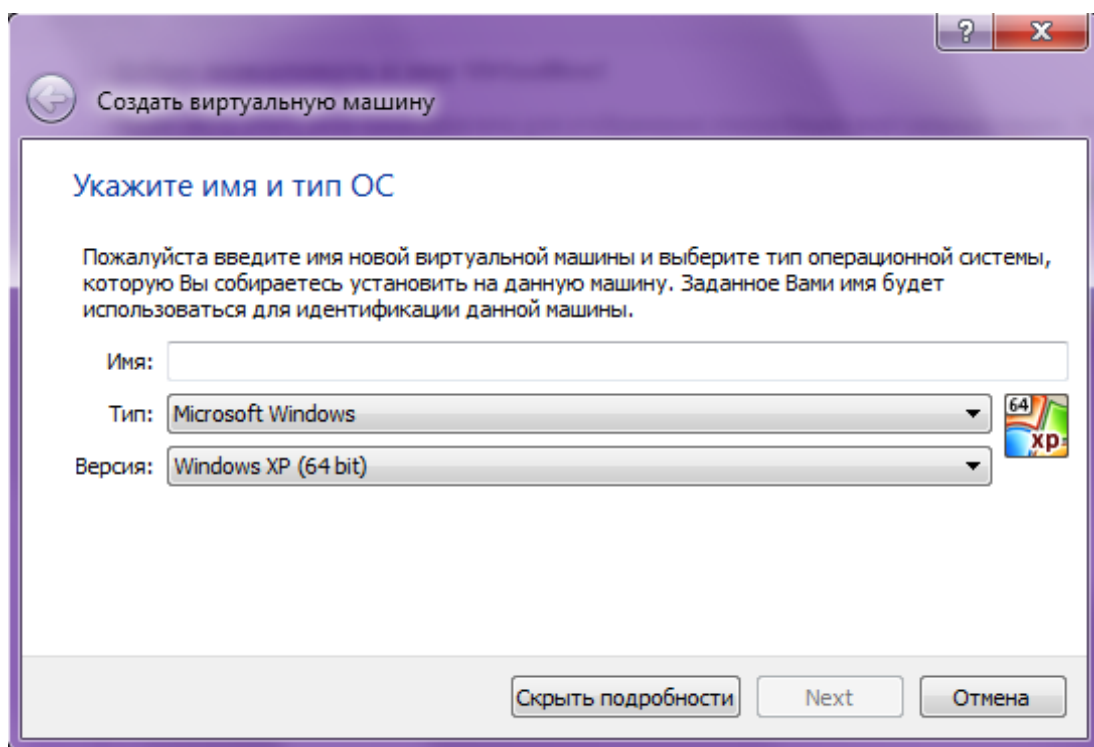


Рисунок 2 – Экранная форма «Мастер создания новой машины»

В зависимости от выбора ОС VirtualBox устанавливает определенные настройки виртуальной машины, необходимые для гостевой ОС. Это особенно важно для 64-битных «гостей». Поэтому рекомендуется всегда устанавливать правильное значение этой настройки.

Следующий параметр – объем оперативной памяти (RAM) который необходим виртуальной машине (рис. 3). При каждом запуске виртуальной машины, VirtualBox будет выделять указанный размер памяти рабочего компьютера и предоставлять ее гостю.

Будьте внимательны на этом этапе! Выделенная память не будет доступна реальной ОС, пока выполняется виртуальная машина, поэтому не указывайте размер больше, чем необходимо.

Так, например, для ОС Windows XP рекомендуемый минимум 512 МБ, но при большем размере оперативной памяти работа системы заметно ускоряется. Для запуска сложных графических приложений в виртуальной ма-

шине может понадобиться больший объем памяти.

Как и прочие настройки, данную можно сменить уже и после создания виртуальной машины.

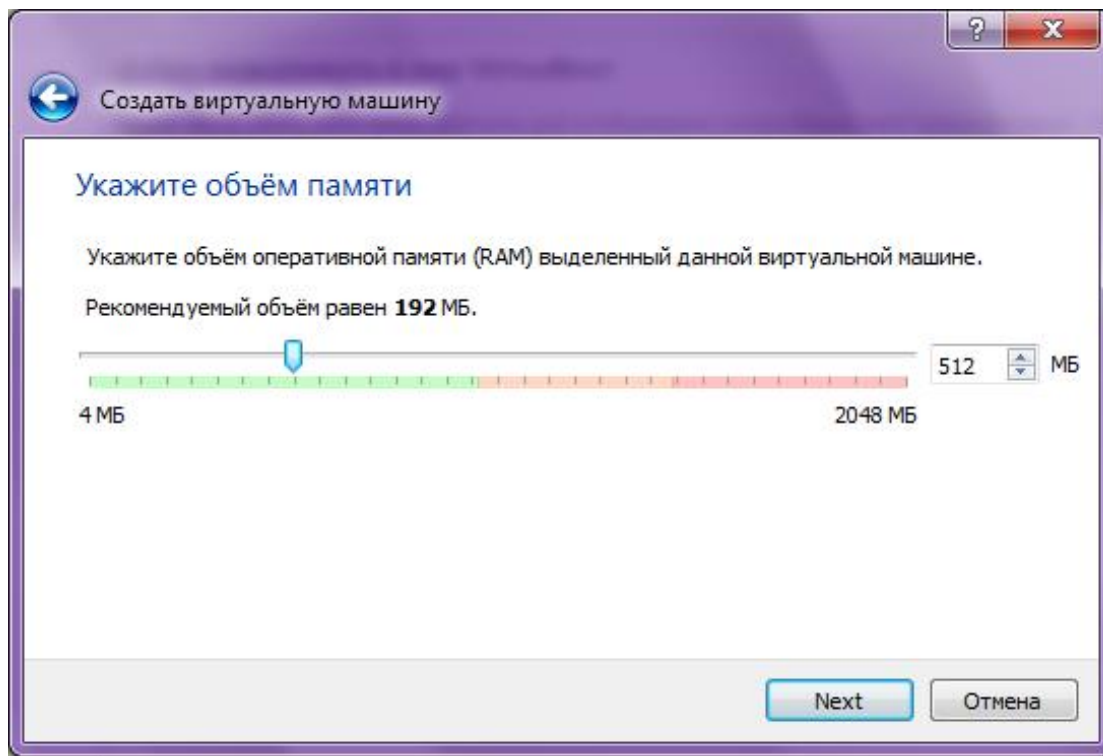


Рисунок 3 – Форма «Выделение ОП»

Далее необходимо указать объем виртуального жесткого диска для виртуальной машины. Для этого существует несколько способов, но наиболее востребованным является метод использования файла образа на "реальном" жестком диске. Его содержимое отображается в виртуальной машине как настоящий диск (рис.4).

Можно создать файл образа или выбрать его из уже существующих. Причем, образы дисков могут быть разделены между различными виртуальными машинами, и при удалении виртуальной машины необходимо сохранить образ или сделать его копию.

В диалоге этой формы можно указать новый виртуальный диск или выбрать из выпадающего списка созданные ранее виртуальные диски, которые не привязаны к другим виртуальным машинам.

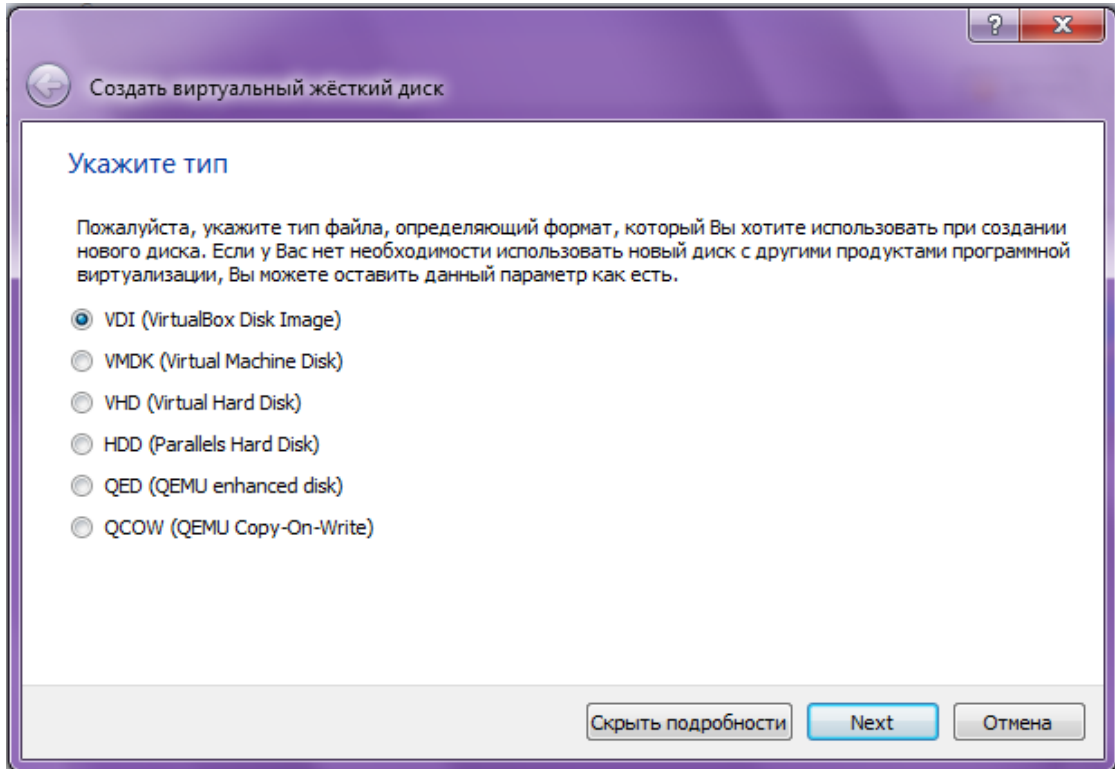


Рисунок 4 – Форма «Мастер создания виртуального диска»

Для создания нового виртуального диска требуется выбрать "New/Новый", осуществив переход в другое окно (рис.5).

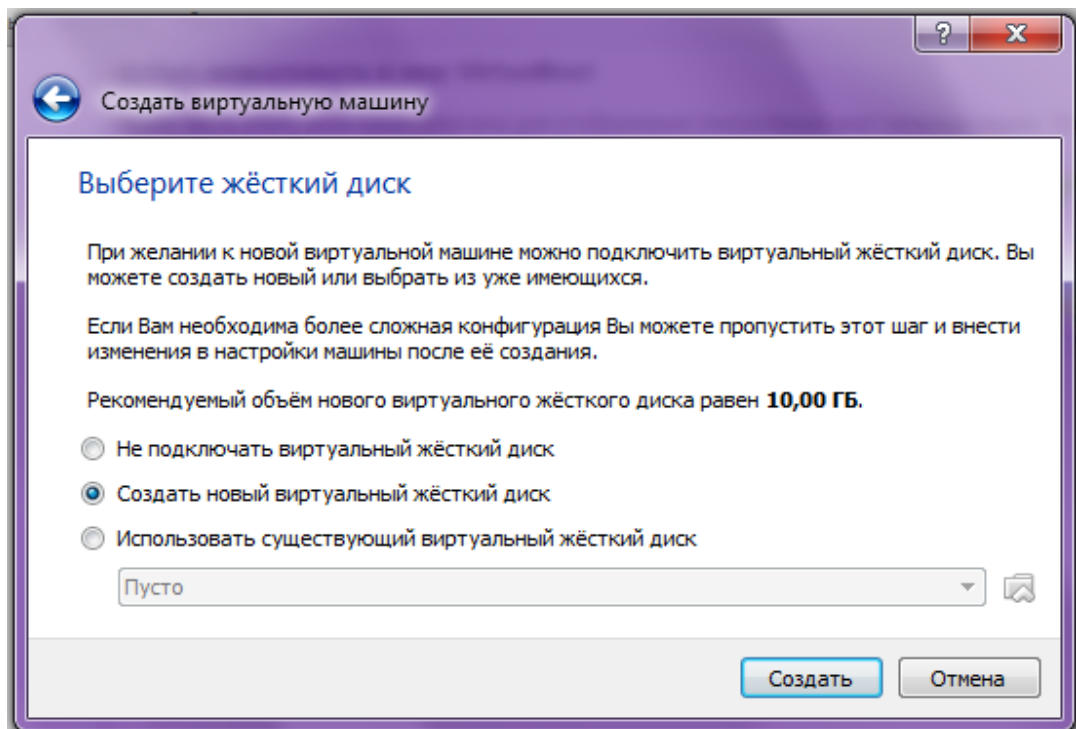


Рисунок 5 – Форма «Мастер создания нового виртуального диска»

В VirtualBox существует два типа файлов образов:

1. Динамически расширяющийся образ, который увеличивается в размере, только тогда, когда гость записывает данные на этот виртуальный диск. Поэтому первоначальный размер файла образа будет небольшим, впоследствии увеличивающимся до установленного размера по мере записи на него данных. Выбрав динамический виртуальный жёсткий диск, можно создать диск минимального объёма, расширяющийся при необходимости до некоторого предела (рис. 6).

2. Образ фиксированного размера, который сразу занимает на реальном диске указанный размер, даже если он использован не полностью. Он занимает больше места на диске, но работа с ним немного быстрее, чем с динамическим.

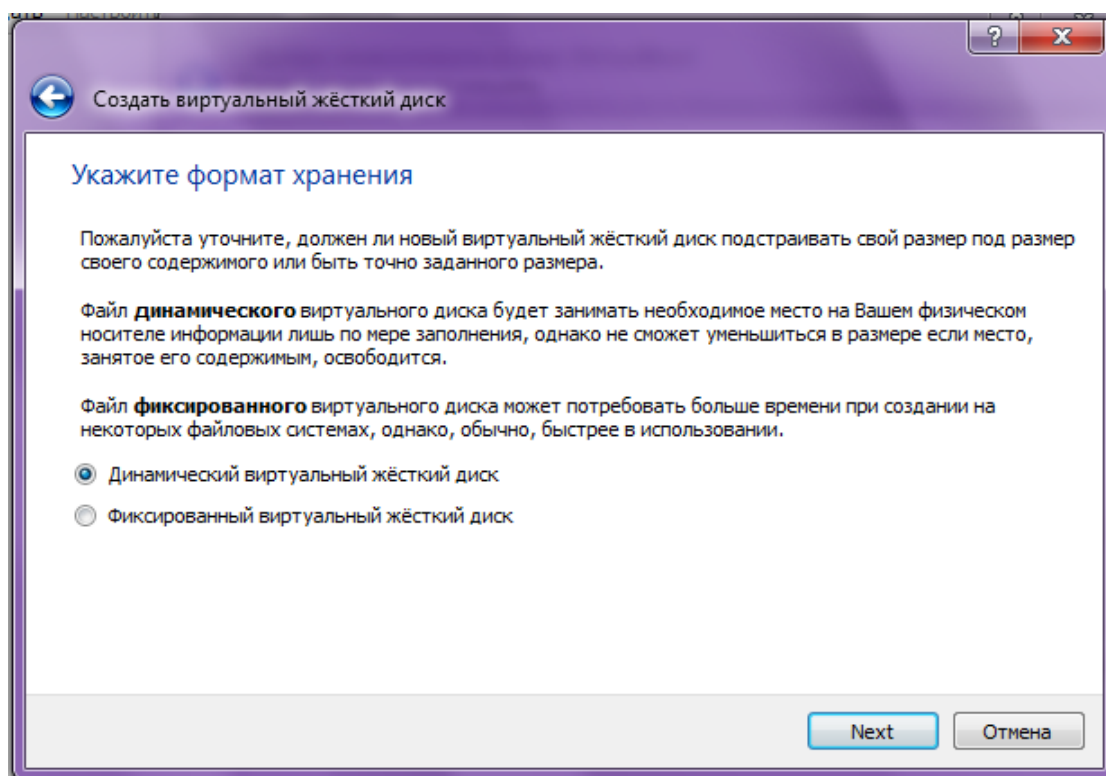


Рисунок 6 – Форма «Мастер создания нового виртуального диска»

Для предотвращения заполнения физического жесткого диска,

VirtualBox ограничивает размер файла образа. Тем не менее, размер образа должен быть достаточно большим, чтобы вместить ОС и приложения, которые планируется установить – для современных семейств Windows и Linux понадобится несколько гигабайтов.

На следующем шаге нужно указать имя диска и его максимальный размер. Размер жёсткого диска нужно выбирать аналогично выбору размера оперативной памяти – в соответствии с имеющимися возможностями реальной машины и потребностями виртуальной. Для Windows XP рекомендуемый размер жёсткого диска 15-20 гигабайтов (рис. 7).

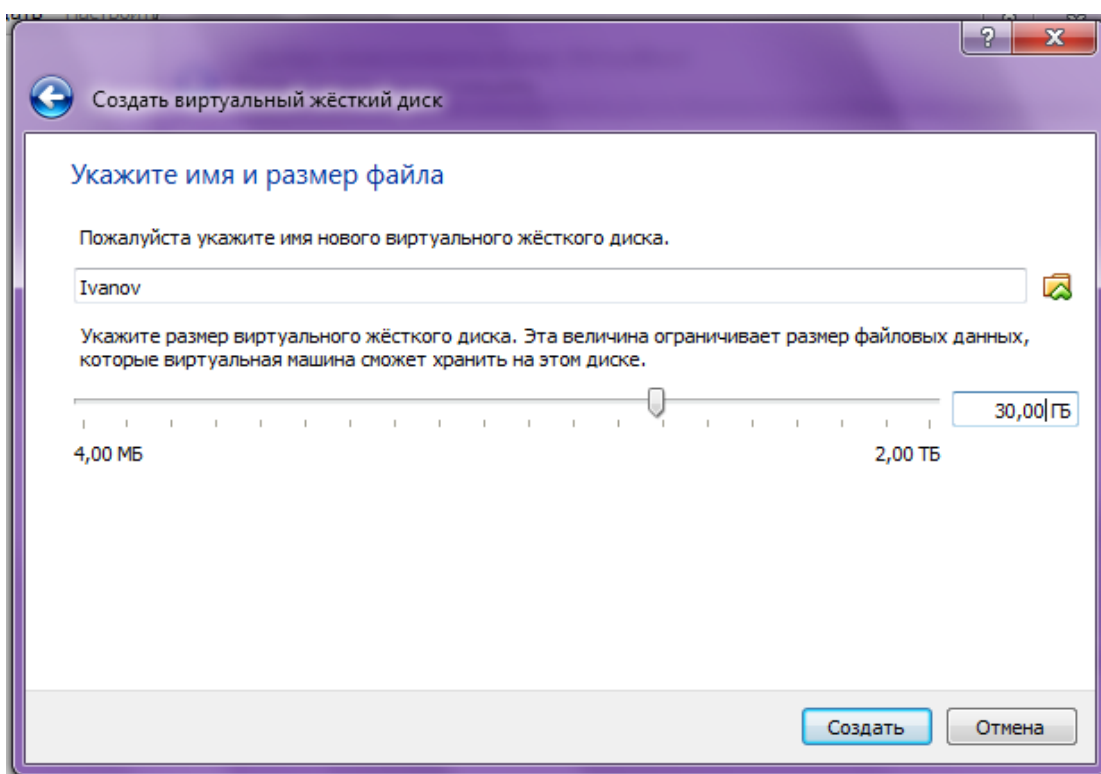


Рисунок 7 – Форма «Мастер создания нового виртуального диска»

После клика на "Создать" новую виртуальную машину можно увидеть в списке в левой части главного окна (рис.8).

В случае необходимости, можно настроить параметры машины в окне менеджера (настраиваемая машина должна быть выключена).

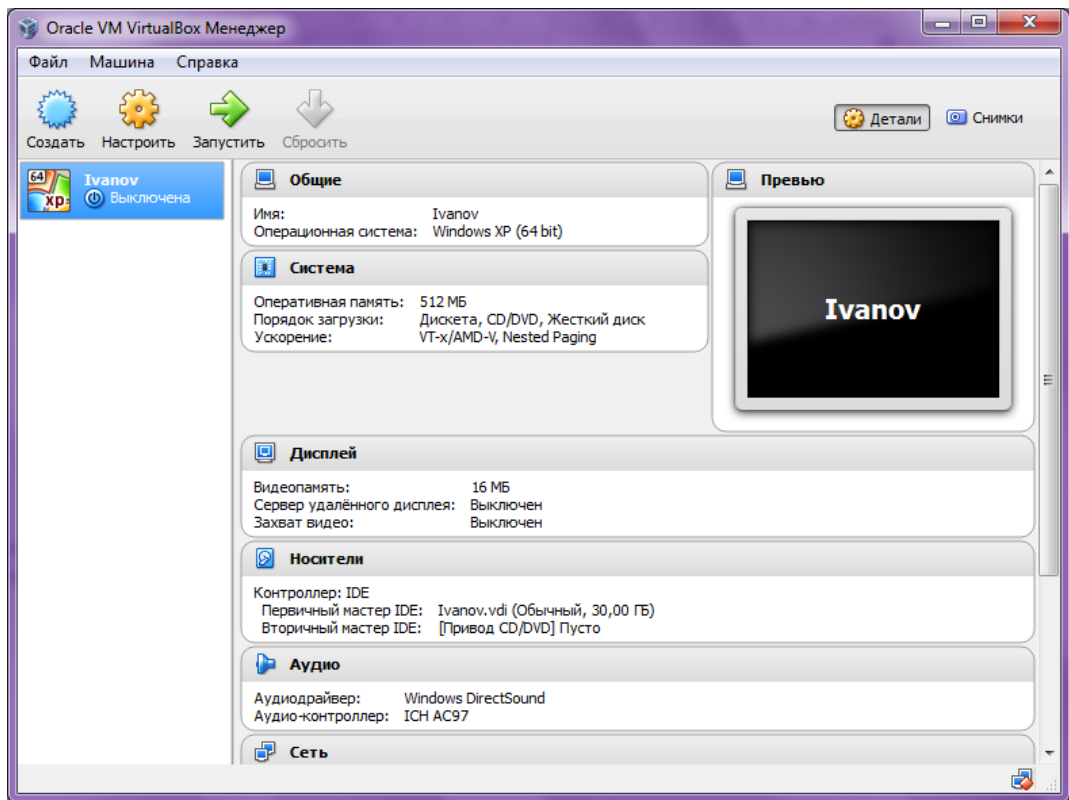


Рисунок 8 – Форма «Главное окно»

Запуск созданной виртуальной машины откроет окна на рис. 9 – 10.

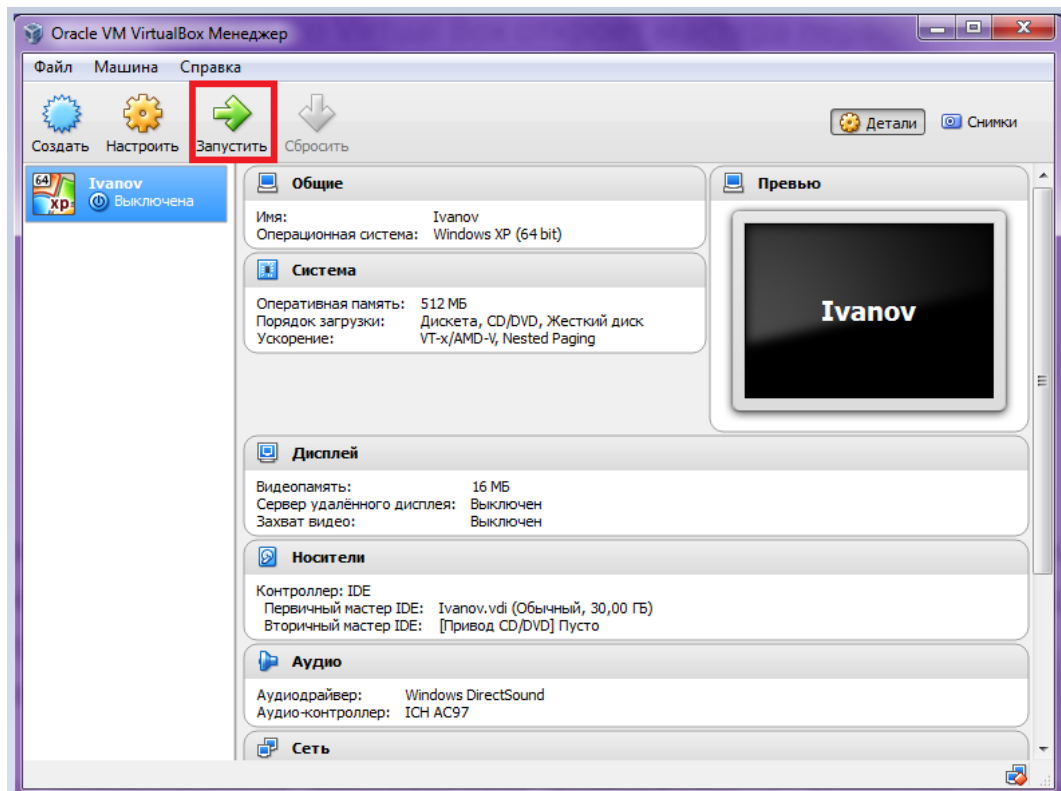


Рисунок 9 – «Главное окно»

Этот этап необходим, поскольку задана только аппаратная конфигурация машины и все её жёсткие диски пусты.

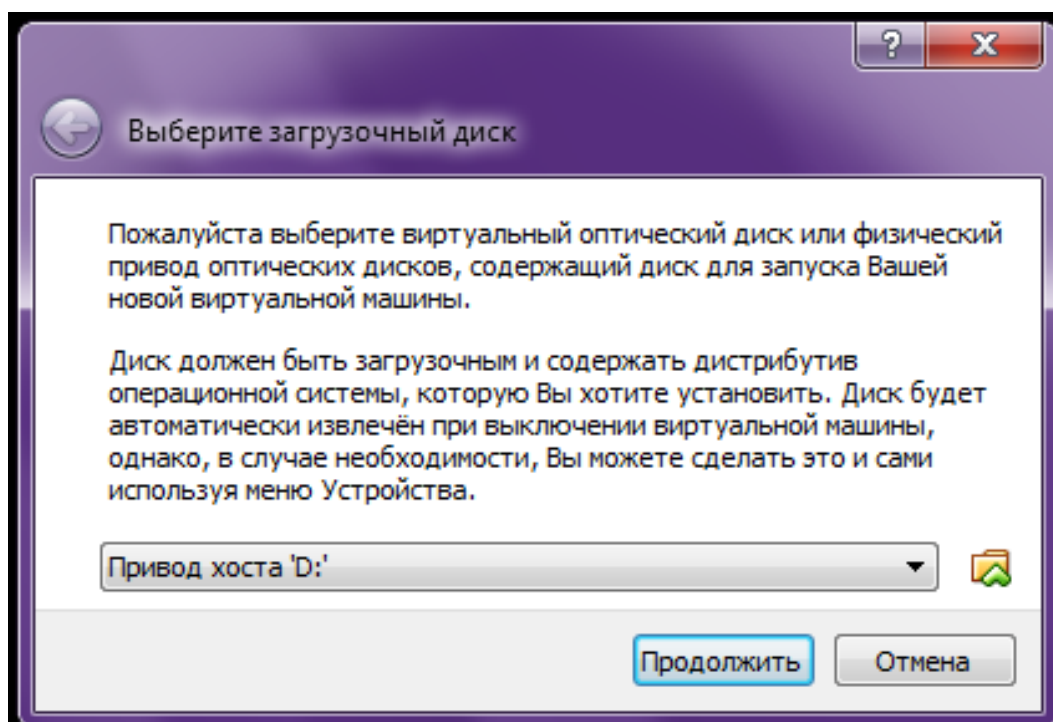


Рисунок 10 – Форма «Мастер первого запуска»

Данный мастер предлагает выбрать образ загрузочного или установочного диска. Можно взять один из имеющихся в списке или указать путь к образу (кнопка в виде папки со стрелочкой справа от списка).

Выполнив установку выбранной операционной системы, получаем полноценный виртуальный образ компьютера, с которым можно экспериментировать и не навредив его реальному прототипу.

Связь файловых систем

Для нормальной последующей работы виртуальной машины нужно установить дополнения гостевой ОС. Для этого в меню виртуальной машины (рисунок 11) нужно выбрать пункт «Устройства – Подключить образ диска с дополнениями гостевой ОС».

После монтирования автоматически должна начаться установка, если этого не произошло, запустите файл `CD:\VBoxWindowsAdditions.exe`

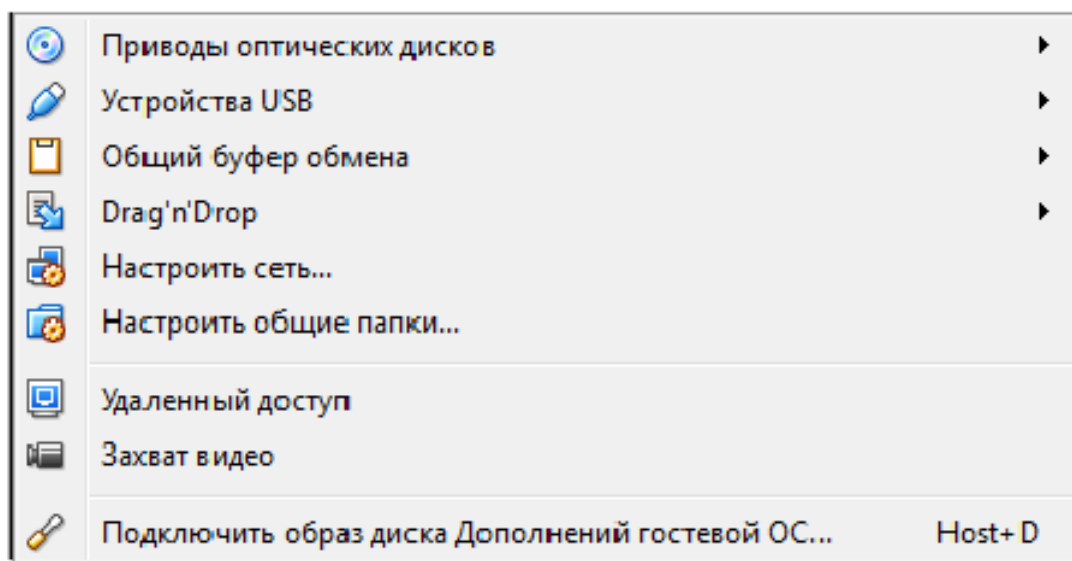


Рисунок – 11 Форма «Устройства»

Следует согласиться со всеми предложениями мастера (оставляя все настройки по умолчанию). Операционная система может выдавать предупреждения и сообщения об ошибках во время установки, тем не менее, следует продолжить установку.

После завершения установки, виртуальную машину нужно перезагрузить, если этого не произошло автоматически.

Для передачи в гостевую (виртуальную) ОС каких-либо файлов, например, установки программ, следует настроить общую папку, выбрав в меню машины «Устройства» пункт «Настроить общие папки»:

Выбрав пункт «Папки машины» в списке общих папок, нажмите кнопку в виде папки с зелёным плюсом – откроется окно настройки пути и имени общего каталога.

В выпадающем списке укажите путь к каталогу, расположенному на реальной машине, доступ к которому требуется открыть из гостевой ОС. Введите имя, под которым данный каталог будет виден в виртуальной машине.

Включите опции «Автоподключение» и «Создать постоянную папку».

Завершите подключение папки нажатием кнопки «Ок». Теперь общая

папка создана, доступна в виртуальной ОС и находится в каталоге: \\Vboxsvr\.

Полезно также настроить общий буфер обмена, чтобы было можно копировать и вставлять данные из одной ОС в другую обычной комбинацией «копировать – вставить». Для этого в том же меню «Устройства» нужно выбрать пункт «Общий буфер обмена» и включить режим «Двунаправленный». Он позволит как копировать данные из ОС реальной машины в ОС виртуальной, так и обратно. Значение остальных режимов интуитивно понятно.

Настройка доступа к сети Интернет

Этап 1. Настройка аппаратной конфигурации

По аналогии с реальным компьютером, которому для доступа к сети, нужна сетевая карта и подключённый к ней кабель, виртуальной машине нужен настроенный соответствующим образом сетевой адаптер.

Зайдите в настройки созданной виртуальной машины, перейдите в раздел «Сеть». Здесь нужно задать параметры сетевых адаптеров, которые будут доступны виртуальной машине.

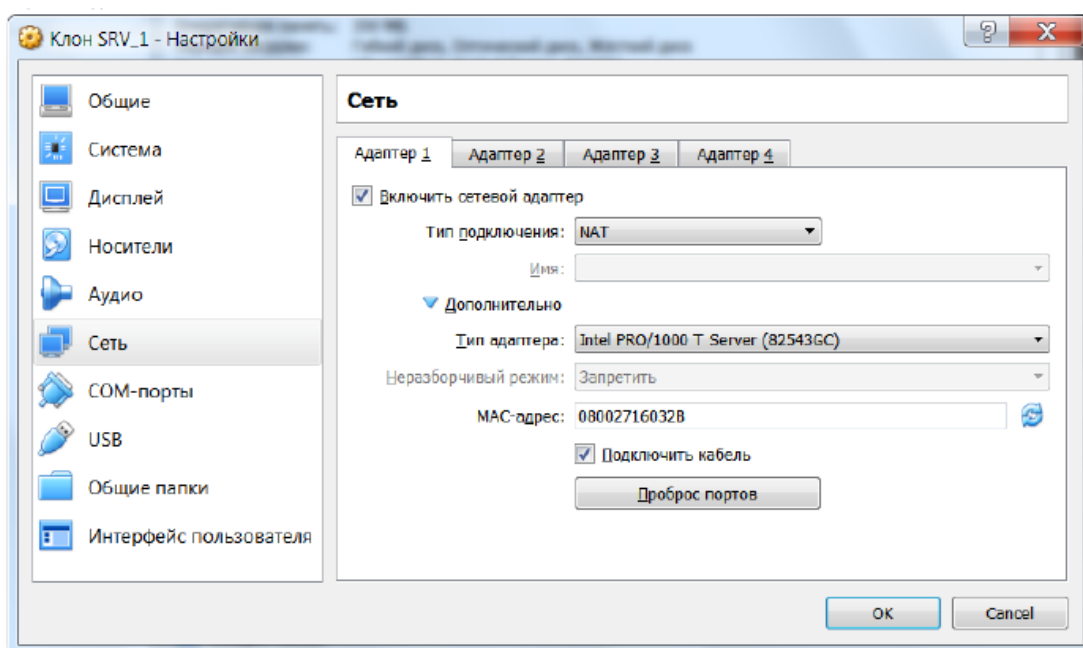


Рисунок 12 – Форма «Настройки»

Оставьте включённым только один адаптер. Установите его параметры в следующие значения:

1. тип подключения – «NAT»;
2. тип адаптера – можно оставить по умолчанию;
3. MAC-адрес – так же можно оставить по умолчанию (стоит помнить, что некоторые проблемы в работе сети могут быть вызваны совпадением MAC-адресов виртуальных машин);
4. кабель подключён – «Да».

Теперь через этот адаптер машина подключена к той же сети, к какой подключён и основной компьютер (компьютер-хост). На этом подготовка аппаратной части виртуальной машины завершена.

Этап 2. Настройка программной конфигурации

Войдите в папку «Мой компьютер – Сетевое окружение», нажав на ссылку «Отобразить сетевые подключения», просмотрите список подключений доступных на данный момент.

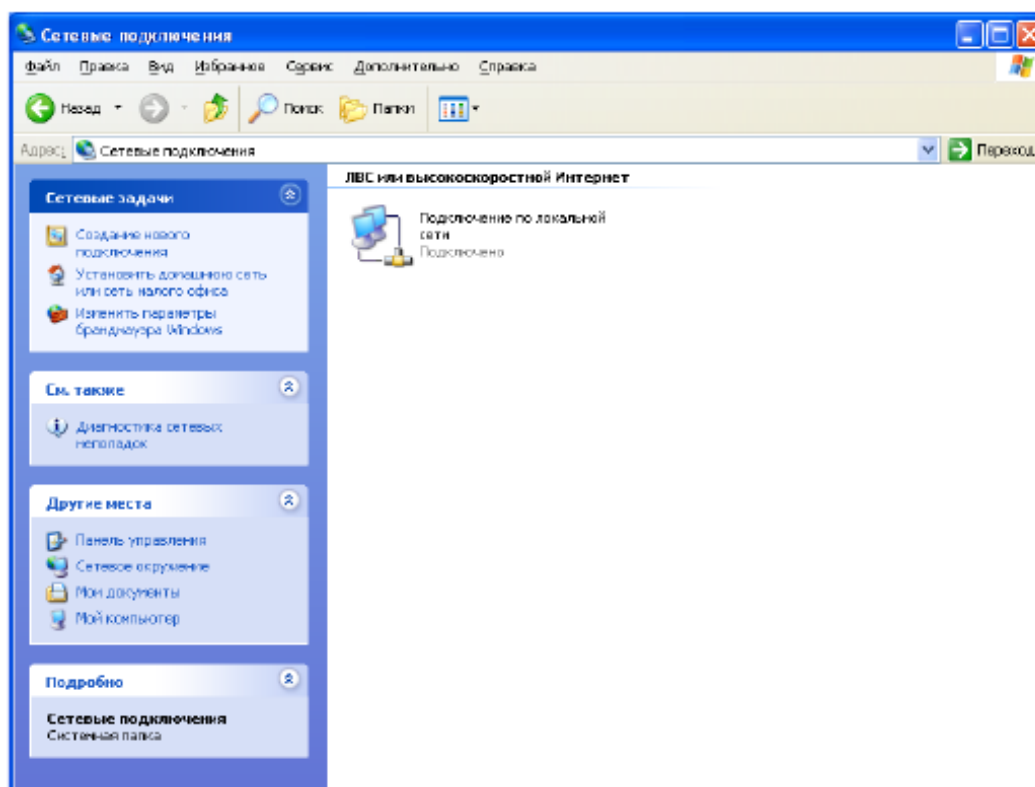


Рисунок – 13 Форма «Сетевые подключения»

На данный момент должно быть только одно подключение по локальной сети. Через контекстное меню откройте свойства этого подключения. Выберите пункт «Протокол Интернета». Установите переключатели в положение «Получить автоматически» – теперь можно вручную задать необходимые параметры подключения к сети.

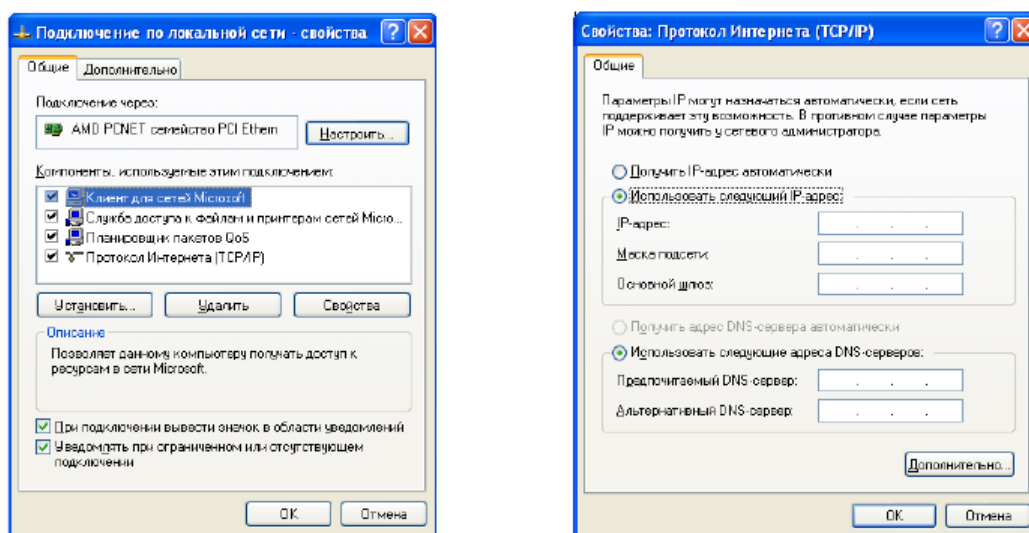


Рисунок – 14 Настройка сетевых подключений

Так же, рекомендуется поставить флажки «При подключении вывести значок в область уведомлений» и «Уведомлять при ограниченном или отсутствующем подключении», что позволит удобнее и проще получать доступ к настройкам сети и к информации о состоянии сетевых подключений.

Этап 3. Настройка прокси-сервера

Доступ устройств локальной сети к Интернету может быть организован разными способами. Один из таких способов – использование прокси-сервера. Прокси-сервер или просто «прокси» – это узел сети, предоставляющий доступ к интернету другим узлам сети. Так же, прокси может выполнять функции фильтрации трафика, ограничения доступа и т. п.

Если в сети доступ в Интернет происходит через прокси, нужно, чтобы локальные компьютеры знали IP-адрес и порт этого сервера. В ОС Windows чтобы настроить прокси- сервер нужно открыть «Свойства обозревателя»

(«Пуск – Панель управления»), перейти на вкладку Подключения.

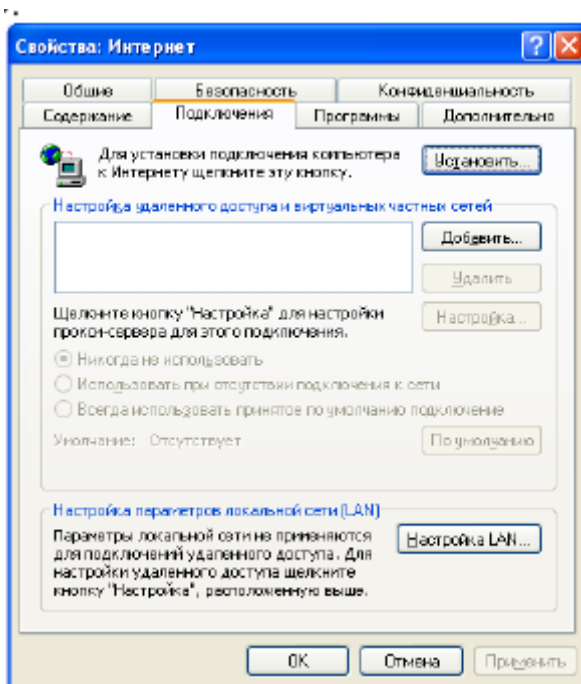


Рисунок – 15 Форма «Свойства обозревателя»

Нажать на кнопку «Настройка LAN» и задать следующие параметры:

1. флажок «Использовать прокси-сервер ...» – установлен;
2. «Адрес» – 62.76.124.3;
3. «Порт» – 8080;
4. флажок «Не использовать прокси-сервер для локальных адресов»

– установлен.

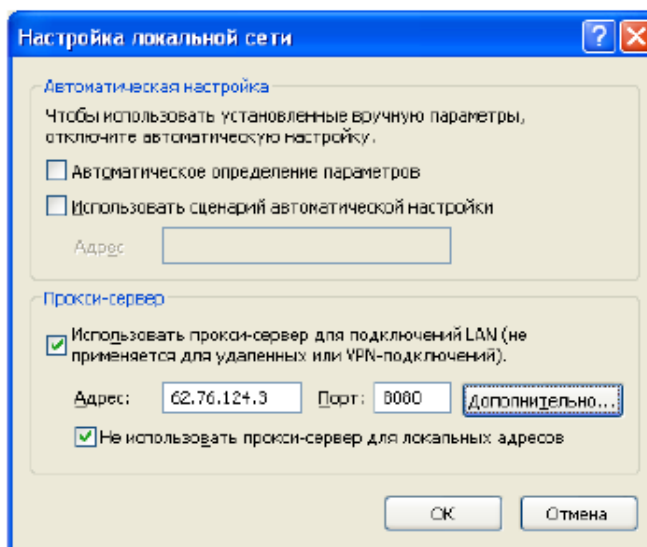


Рисунок – 16 «Настройка локальной сети»

Сохраните сделанные настройки. На этом третий этап завершён. Осталось проверить работу сделанных настроек.

Этап 4. Проверка работоспособности сделанных настроек

Проверку можно выполнить следующими способами:

1. попробовать открыть какую-нибудь страницу интернета в браузере;
2. выполнить команду ping в командной строке.

Для проверки доступности узла сети и получения информации о качестве связи с ним обычно используется команда ping командной строки Windows («Пуск – Все программы – Стандартные – Командная строка» или «Пуск – Выполнить – cmd»).

По команде ping компьютер отправляет серию специальных пакетов на указанный адрес узла сети и ждёт от него ответ. Если сеть работает хорошо, то меньше чем через секунду компьютер получит ответ и выведет информацию о соединении.

Итак, синтаксис команда ping имеет следующий синтаксис:

ping NodeAdress

Здесь NodeAdress – адрес узла, доступность которого проверяется. Адресом узла может быть как его числовой IP-адрес, так и один из символьных адресов. Ниже приведён пример успешного выполнения команды ping (пингования) одного из серверов Яндекса:

```
C:\Users\AdMUNUcmramop>ping ya.ru
```

```
Обмен пакетами с ya.ru [93.158.134.3] с 32 байтами данных;
```

```
Ответ от 93.158.134.3: число байт=32 время=109мс TTL=56 Ответ от 93.158.134.3; число байт=32 время=109мс TTL=56 Ответ от 93.158.134.3; число байт=32 время=109мс TTL=56 Ответ от 93.158.134.3; число байт=32 время=109мс TTL=56
```

```
Статистика Ping для 93.158.134.3;
```

```
Пакетов; отправлено = 4, получено = 4, потеряно = 0 (0% потерь)
```

Приблизительное время приема-передачи в мс;

Минимальное = 109мс, Максимальное = 109 мс, Среднее = 109 мс

ya.ru – символьный адрес пингуемого узла сети, он преобразуется ОС в соответствующий IP-адрес 93.158.134.3;

число байт – информационный вес каждого из 4-х отправленных пакетов; время – время, за которое пакет доходит до пингуемого узла. Это время так же называют задержкой. Обычно, нормальным считается значение этого показателя меньше 200 миллисекунд. Большая разница в значениях задержки за сравнительно короткий промежуток времени. как правило, свидетельствует о наличии проблем в сети.

Выполните проверку двумя способами. Если во время проверки выявились проблемы, перезагрузите виртуальную машину и выполните проверки снова.

Контрольные вопросы

1. Сколько этапов необходимо выполнить для настройки доступа виртуальной машины к сети Интернет? Каковы эти этапы?
2. Поясните термин «виртуальная машина».
3. Какой размер виртуального жесткого диска следует задать при установке образа современной операционной системы семейств Windows и Linux?
4. Перечислите функции прокси-сервера.
5. Какие типы файлов образов существуют в VirtualBox?

Лабораторная работа №2

Тема: Командная строка Windows и командные файлы

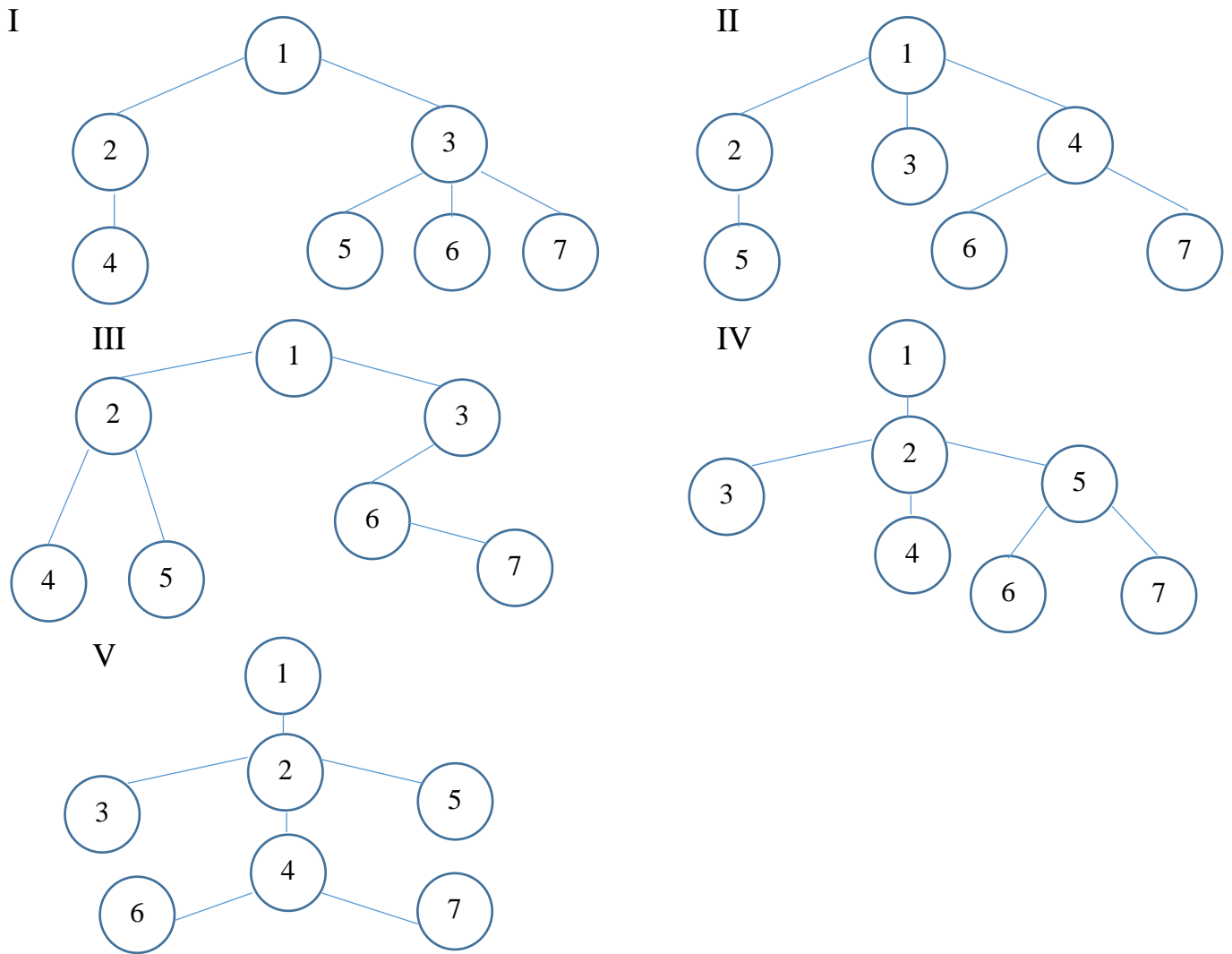
Цель: Развитие навыков работы с командной строкой в операционной системе семейства Windows и умений по созданию командных файлов.

Задание

1. В командной строке ОС Windows выполните следующие действия:

- создайте каталог для лабораторной работы;
- создайте дерево подкаталогов в соответствии со своим вариантом;
- в каталоге 1 создайте файл с любым текстом;
- выполните копирование каталога 3 в 2;
- переместить файлы из каталога 2 в 3;
- скопируйте в каталог 5 три файла jpg и два doc;
- просмотрите список файлов директории 5;
- переименуйте два изображения в wty4.jpg и в k4b.jpg;
- переименуйте один из файлов doc в jer4.doc;
- просмотрите список файлов директории 5;
- скопируйте каталог 5 в 4;
- удалите файлы, имя которых заканчивается на "4" из каталога 5;
- просмотрите список файлов директории 5;
- удалите из каталога 4 все файлы, содержащие в имени "4";
- просмотрите список файлов директории 7;
- удалите из каталога 4 все оставшиеся изображения;
- удалите каталог 6.

Варианты деревьев:



2. Создать командный файл, выполняющий следующие действия:

Вариант 1

Копирование на USB-носитель файла, заданного как параметр при запуске командного файла;

Переименование скопированного на предыдущем шаге файла, новое имя должно быть задано при запуске командного файла как параметр;

Удаление копии файла с диска С.

Вариант 2

Удаление всех файлов с папки флэшки, имя папки задать как параметр при запуске файла;

Копирование файлов из каталога DN, имеющих расширение .exe;

Переименование всех файлов в указанной папке таким образом, чтобы они начинались на букву «а»;

Вывод на экран содержимого файла, имя которого задано как параметр.

Вариант 3

Создание директории в корневом каталоге флэшки, имя задать как параметр;

Перенос в созданную директорию всех файлов из папки Windows с расширением .doc, с изменением расширения на .txt;

Удаление из этой директории всех файлов, начинающихся на букву «с»;

Открыть для редактирования документ, имя которого задается как второй параметр.

Вариант 4

Удаление с USB-носителя директории, имя которой задано как первый параметр;

Создание на USB-носителе файла, имя которого задано как второй параметр;

Создание двух копий этого файла в корневом каталоге жесткого диска, с именами, заданными как параметры;

Удаление с флэшки файла, созданного на втором шаге.

Вариант 5

Удаление из директории, имя которой задано как параметр, всех файлов, имеющих расширение .bak;

Просмотр файл с именем, заданным как параметр;

Копирование этого файла USB-носитель во вновь созданную директорию с именем, заданным как второй параметр.

Вариант 6

Создание текстового файла в корневом каталоге с именем, заданным как параметр;

Копирование из директории «Мои документы» на USB-носитель всех файлов с расширением .doc и .txt;

Удаление с USB-носителя всех файлов, начинающихся на букву, заданную как второй параметр;

Редактирование документа, созданного на первом шаге.

3. Написать диалоговый командный файл, выполняющий следующие действия по выбору пользователя:

а) Запуск антивирусной программы;

б) Запуск программы-оболочки;

в) Запуск одного из редакторов текста;

г) Запуск среды программирования по выбору;

Указать время ожидания и ответ, принимаемый по умолчанию

Технология выполнения работы

В целях безопасности обязательно создавайте командные файлы (bat-файлы) в отдельной директории!

Командная строка

Интерфейс командной строки (англ. Command line interface, CLI) – разновидность текстового интерфейса (CUI) между человеком и компьютером, в котором инструкции компьютеру даются в основном путём ввода с клавиатуры текстовых строк (команд). Также известен под названием консоль.

Командная строка встречается во многих различных программах, например: играх (здесь консоль используется сначала разработчиками для тестирования, а потом игроками для обхода багов, неисправленных разработчиками, читов и просто получения большего удовольствия от игрового процесса; есть игры, в которых консоли являются официальной и полноценной частью интерфейса: Quake и др.), сложных графических процессорах, системах автоматизированного проектирования (AutoCad), консольных браузерах и ряде других программ.

Главными преимуществами командной строки являются:

1. низкие требования к ресурсам, необходимым для работы самой консоли;
2. простота использования: в отличие от графического интерфейса, в командной строке доступны сразу все или почти все команды (действия), интерфейс не перегружен множеством элементов, с которыми нужно взаимодействовать в определённом порядке, чтобы дойти наконец до элементов, выполняющих нужные действия.

А главными недостатками:

1. необходимость знать команды, выполняющие требуемые действия;
2. непривычность способа взаимодействия с компьютером;
3. низкая скорость набора (слабое владение клавиатурой).

Но часто используемые команды легко и быстро запоминаются, а неизвестные команды или параметры можно найти в справочнике. Привыкнуть действовать в консоли не так уж трудно, особенно когда ощущаешь на деле её преимущества и постепенно начинаешь выполнять некоторые действия в командной строке. А для освоения клавиатуры есть много программ-тренажёров, обучающих правильному скоростному набору.

В операционных системах тоже есть командная строка. Собственно, первые операционные системы имели именно командный интерфейс, а операционная система Windows изначально была «оболочкой Windows» над Дисковой Операционной Системой и лишь через несколько версий включила DOS в себя и действительно стала операционной системой.

Преимущества командной строки в операционных системах следующие:

1. для командной строки возможно создание сценариев автоматизации и пакетных файлов, т.е. выполнение одной или нескольких команд без вмешательства пользователя;

2. консоль предоставляет возможность управления данными и файлами. Преимущества командной строки становятся очевидны, когда требуется выполнять однотипные операции над множеством объектов;

3. администрирование компьютера. Быстрое получение текущей информации сокращает время диагностики компьютера;

4. администрирование сети. Многие команды администрирования сети не имеют графических эквивалентов (например – команда ping, pathping, tracert);

5. небольшой расход памяти по сравнению с системой меню.

В разных операционных системах консоль называется по-разному, к тому же, могут существовать различные программы, предоставляющие «расширенную консоль», т.е. расширяющие набор команд встроенной командной строки или какие-либо другие улучшения.

Командные интерфейсы обычно используются на серверах, на которых может не устанавливаться графическая оболочка серверной ОС, т.к. с одной стороны, при этом уменьшается нагрузка на железо сервера, а с другой, обращаясь к серверной ОС будет достаточно квалифицированный персонал.

В ОС Windows командную строку можно вызвать одним из следующих способов:

1. Пуск -> Все программы -> Стандартные -> Командная строка.
2. Пуск -> Выполнить и введите имя программы cmd.exe.

В консоли Windows действует мышь и операции копировать-вставить, но выполняются они из меню окна или контекстного меню.

Маска – это некоторое выражение, задающее множество значений. Проще говоря, в команде копирования файла можно прописать все имена файлов с нужным расширением (a.txt, b.txt, ...), а можно задать тот же набор имён файлов с помощью маски (*.txt): * – заменяет любой набор символов; ? – заменяет один любой символ. Примеры: *.* ; *.TXT ; A*.DB ; A???.TXT ; FILE.?T?

Основные команды, необходимые для работы можно найти, перейдя по ссылке:

http://www.junior.ru/wwwexam/opsys/msdos_k.htm

Перед использованием команды нужно посмотреть её описание в справочнике.

BAT-файлы

Пакетный файл (англ. batch file) – текстовый файл в MS-DOS, OS/2 или Windows, содержащий последовательность команд, предназначенных для исполнения командным интерпретатором. После запуска пакетного файла, программа-интерпретатор (как правило, COMMAND.COM или cmd.exe) читает его строка за строкой и последовательно исполняет команды. Пакетный файл – аналог скриптовых файлов командной строки (shell script) в Unix-подобных операционных системах.

Пакетные файлы позволяют создавать сценарии для командной строки и представляют собой текстовый документ, содержащий команды консоли плюс некоторые другие. Таким образом, постоянно повторяющиеся последовательности команд можно описать в bat-файлах и использовать уже их, не набирая снова всю последовательность команд в консоли. К тому же, bat-файл можно запустить из планировщика задач.

Полезные ссылки

http://www.colorpilot.ru/command_line.html

<http://cmd4win.ru/>

http://forum.xakep.ru/m_1512627/tm.htm

Лабораторная работа №3

Тема: Создание, установка и запуск виртуальной машины для образа операционной системы Ubuntu.

Цель: Получение практических навыков работы с программным продуктом OracleVirtualBox виртуализации для операционной системы семейства Linux.

Задание

1. Создайте виртуальную машину
2. Установите образ Ubuntu
3. Обеспечьте подключение к интернету

Технология выполнения работы

Ubuntu – дистрибутив GNU/Linux, основанный на Debian GNU/Linux. Основным разработчиком и спонсором является компания Canonical. В настоящее время проект активно развивается и поддерживается свободным сообществом[

Создание виртуальной машины

Откройте VirtualBox, нажмите «Создать», выберите тип системы и введите имя:

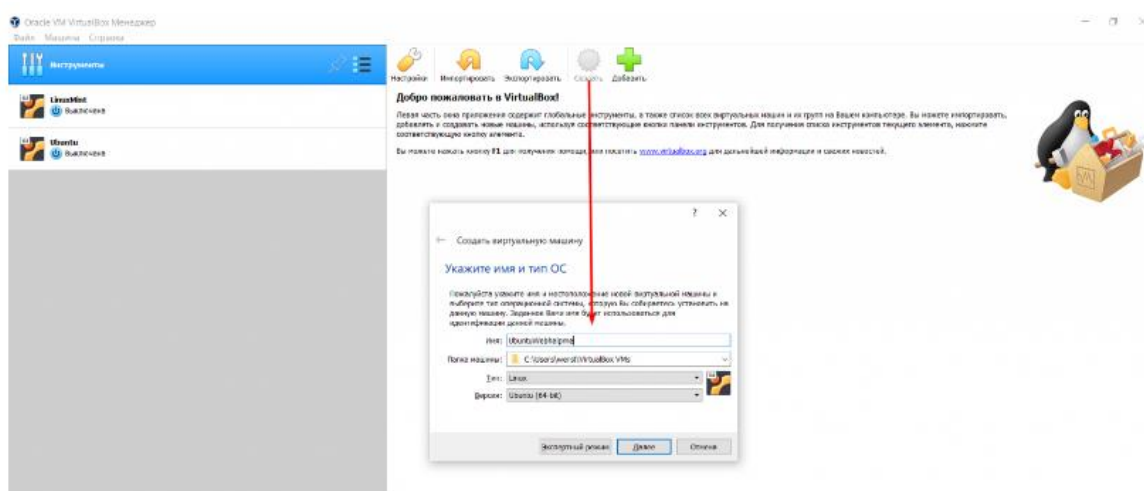


Рисунок 17 – Форма «Создание виртуальной машины»

Выделите ОЗУ для виртуальной машины, при этом не забывайте, что нужно оставить память для вашей основной ОС:

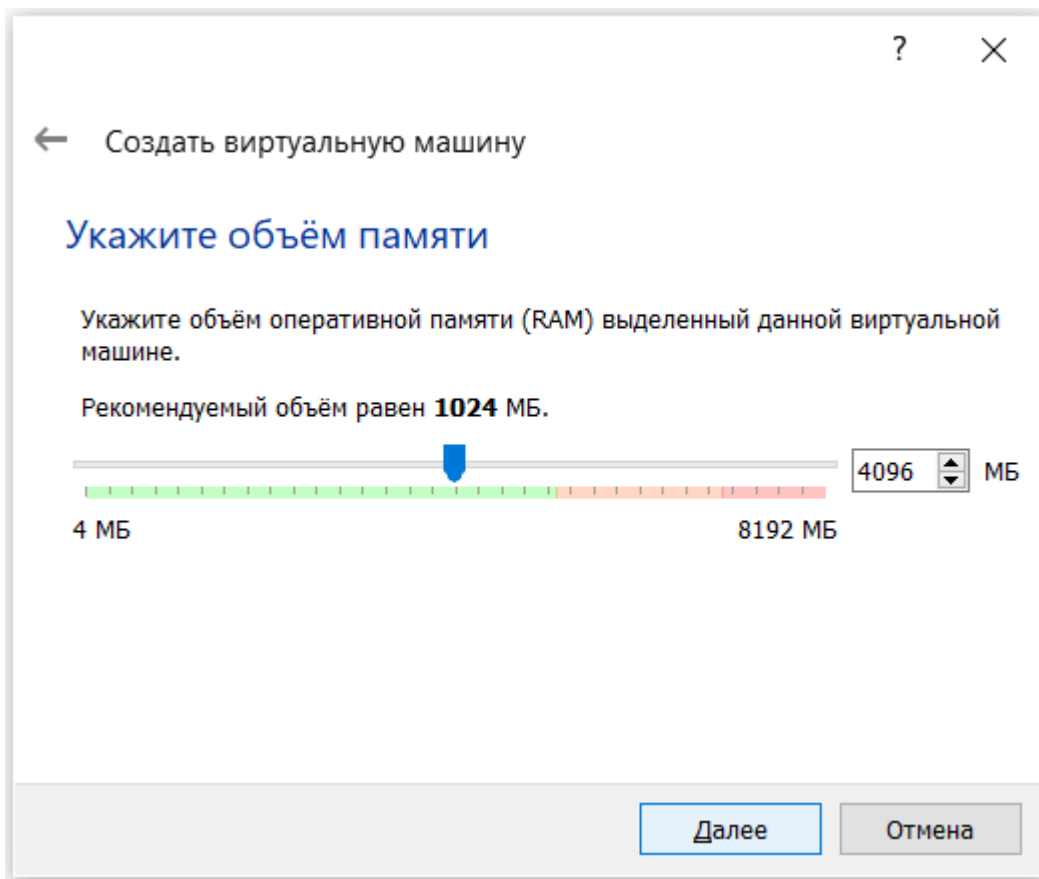


Рисунок 18 – Форма «Объем памяти»

Выберите «Создать новый виртуальный жесткий диск», затем - VHD (Virtual Hard Disk), нажмите Далее. На следующем этапе выберите «Динамический виртуальный жесткий диск» и нажмите Далее.

Укажите размер виртуального диска для вашей системы: объем в 20 гигабайтов является минимальным, но при возможности лучше выделить 40 гигабайтов, что в дальнейшем позволит выполнять установку различных программ (рис. 19).

Далее необходимо выполнить еще несколько настроек, чтобы можно было приступить к непосредственной установке операционной системы.

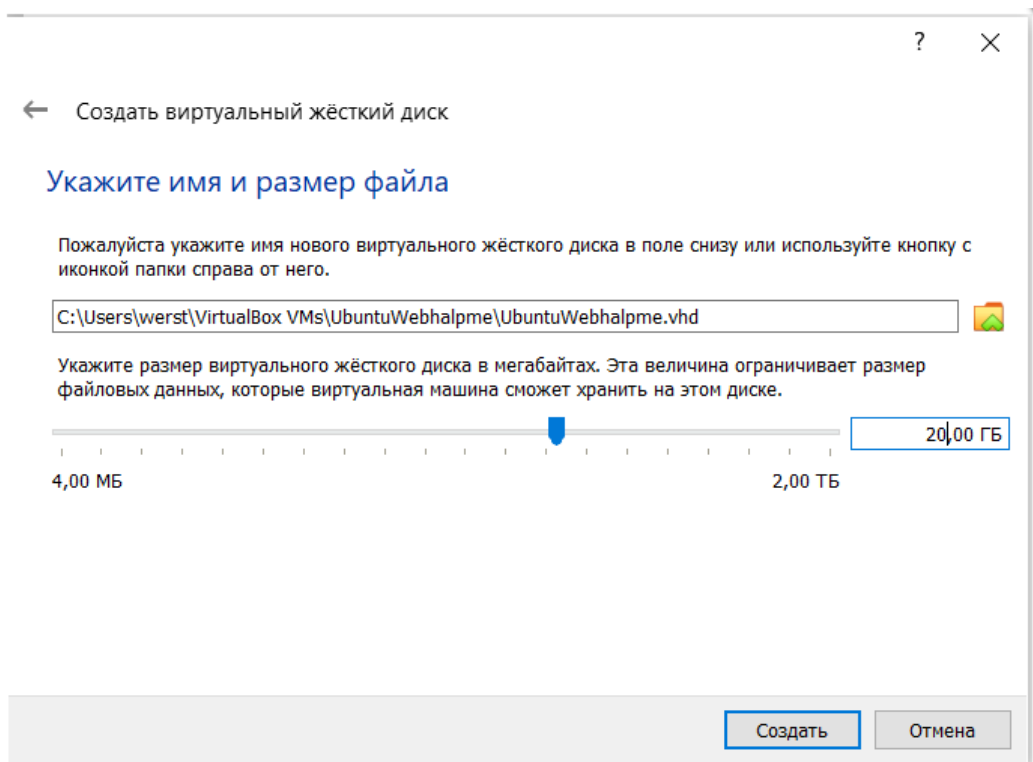


Рисунок 19 – Форма «Создание виртуальной машины»

Осуществите выбор операционной системы (рис. 20), нажмите «Настроить»:

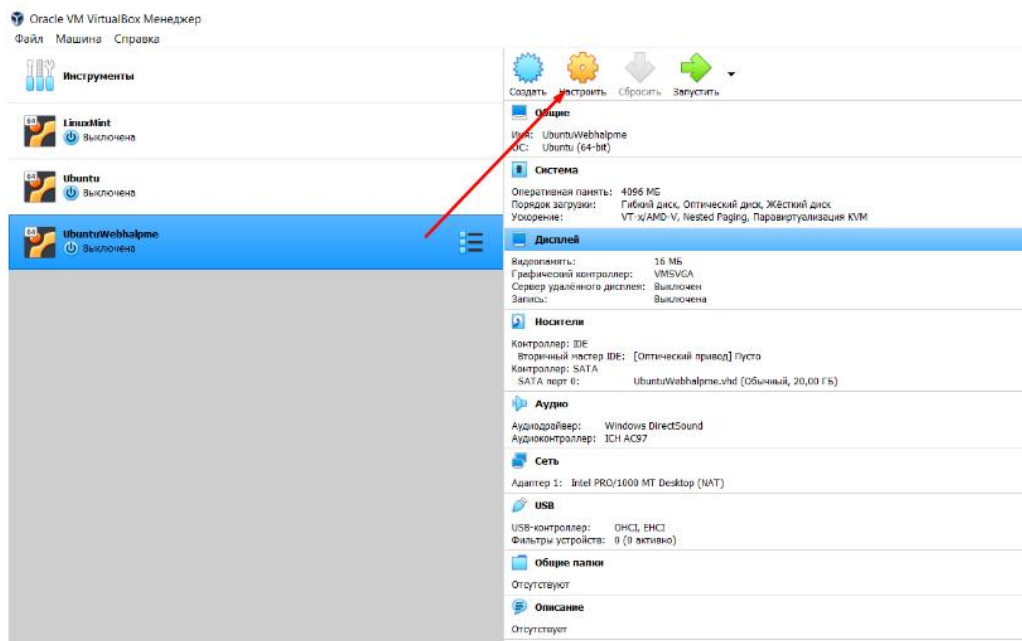


Рисунок 19 – Форма «Выбор образа ОС»

В разделе «Система», вкладка «Процессор» (рис.20) выделите количе-

ство ядер вашего процессора (если у процессора всего 2 ядра, выделите 1 ядро, если всего 4 ядра, выделите 2 и т.д.):

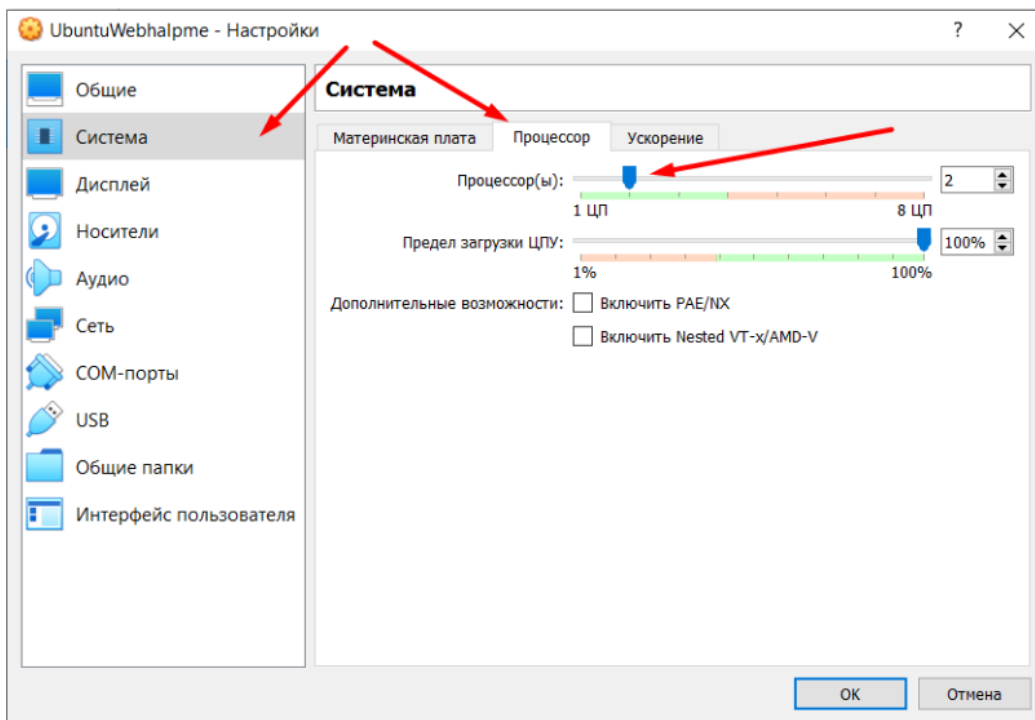


Рисунок 20 – Форма «Настройки»

В разделе «Дисплей»: включите 3D-ускорение и перетяните «Видеопамять» на максимум:

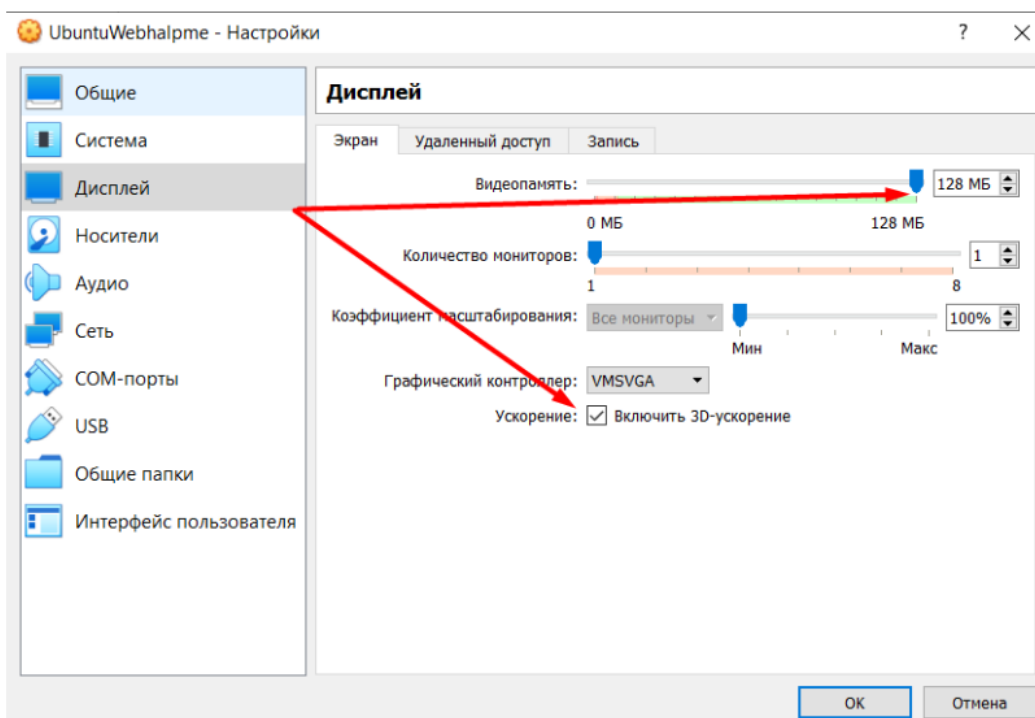


Рисунок 21 – Форма «Настройка видеопамати»

Внизу раздела «Носители» нажмите «Добавить новый контроллер» и выберите «PIIX4 (Default IDE)» (рис.22).

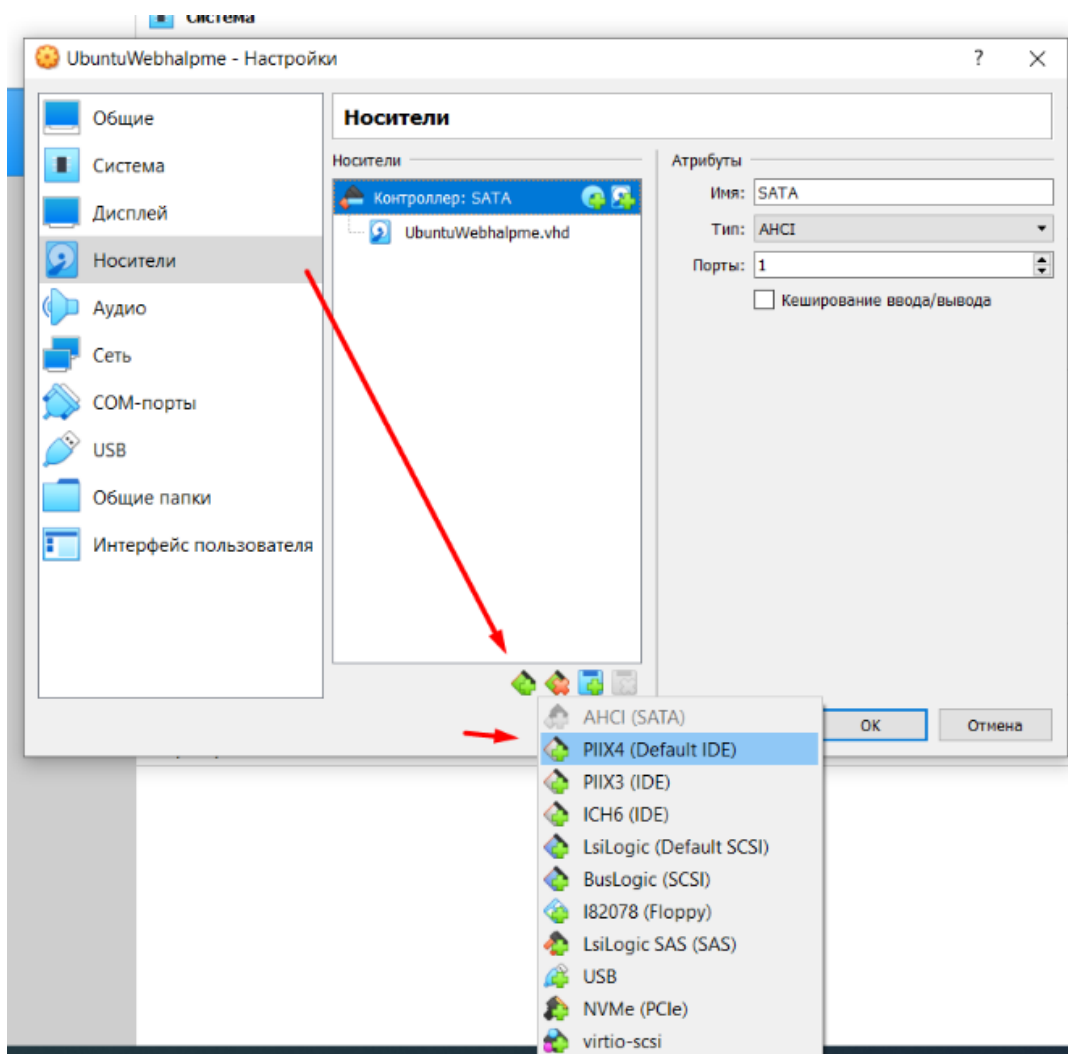


Рисунок 22 – Форма «Новый контроллер»

Добавьте привод оптических дисков на созданном вами контроллере и нажмите «Оставить пустым» (рис.23).

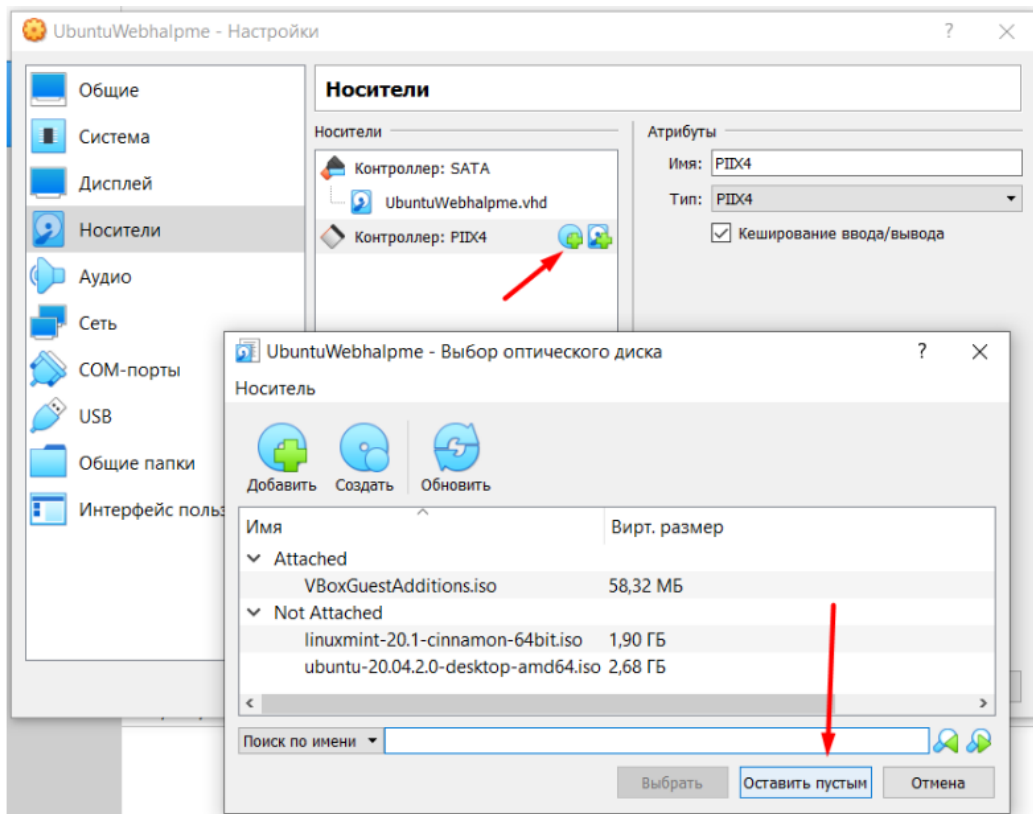


Рисунок 23 – Форма «Привод оптических дисков»

Нажмите на привод «Пусто», затем «Оптический привод» нужно установить «Вторичный мастер IDE» (рис. 24).

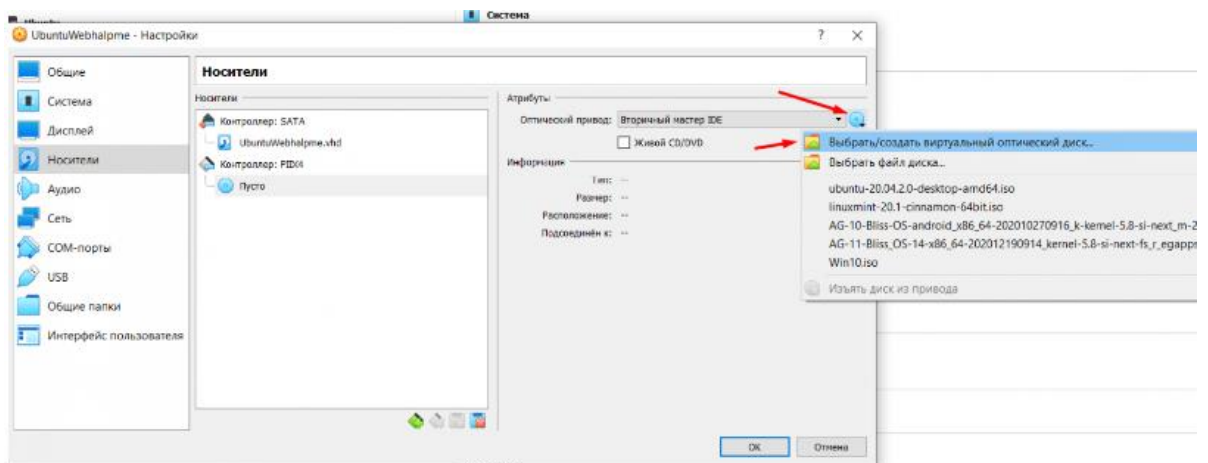


Рисунок 24 – Форма «Настройки»

Нажмите «Добавить» и выберите ваш ISO образ с операционной системой (рис.25).

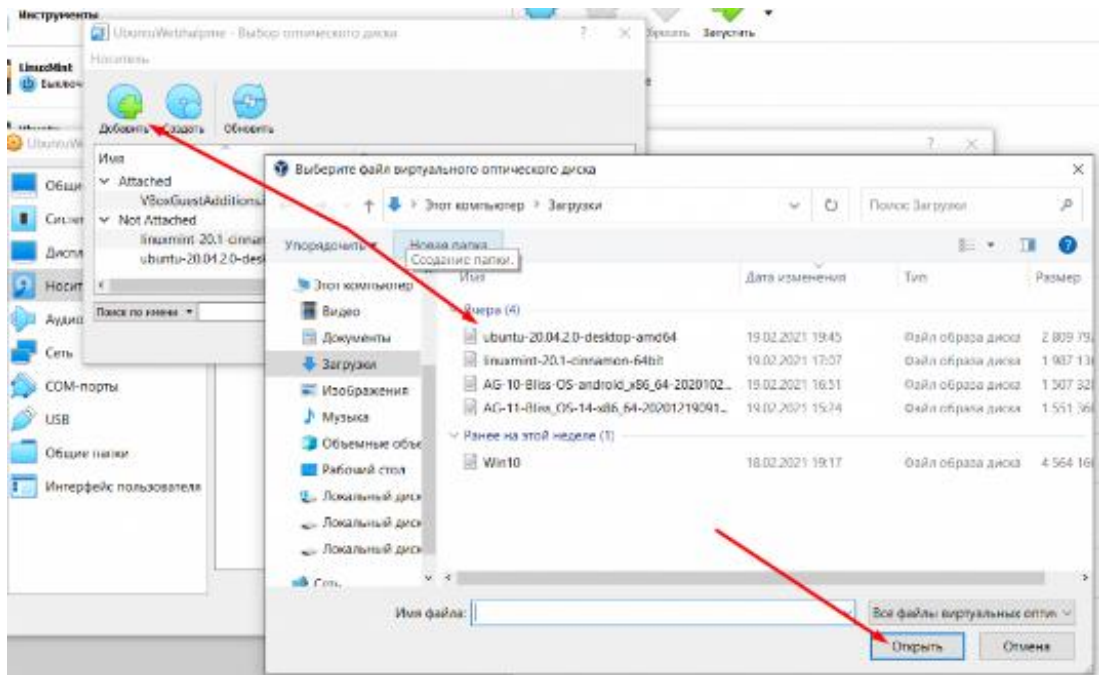


Рисунок 25 – Форма «Настройки»

Осталось лишь выбрать ваш ISO (рис.26).

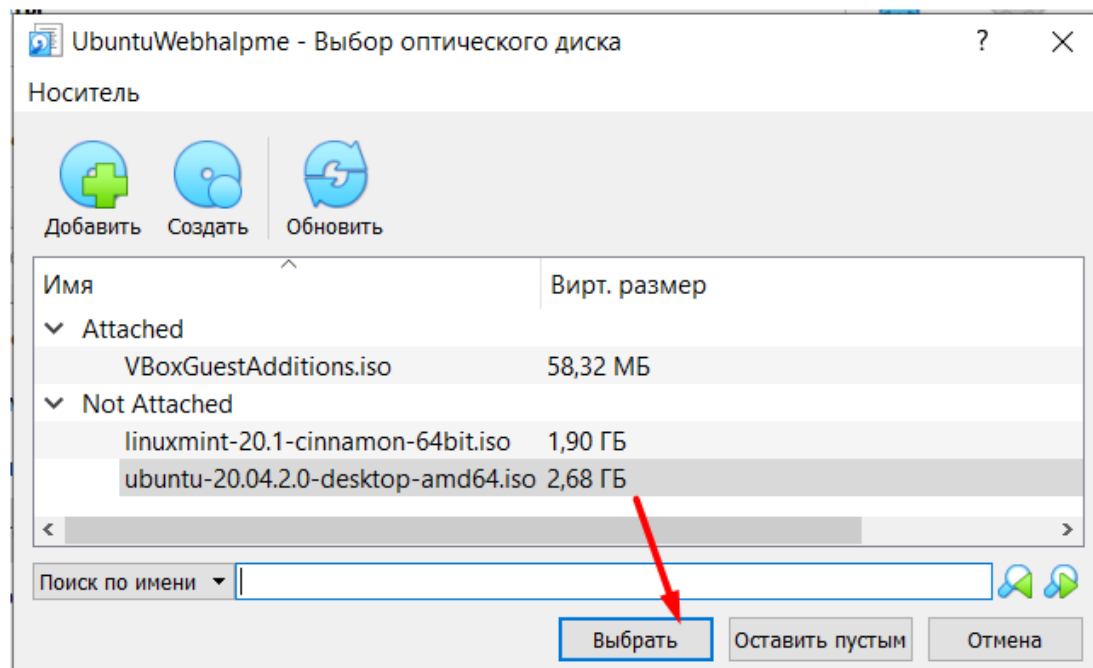


Рисунок 26 – Форма «Настройка ISO»

Нажмите «ОК». На этом настройка аппаратной части завершена.

Установка Ubuntu

В главном меню менеджера Virtual Box выберите, нужную аппаратную конфигурацию и нажмите запустить (рис. 27).

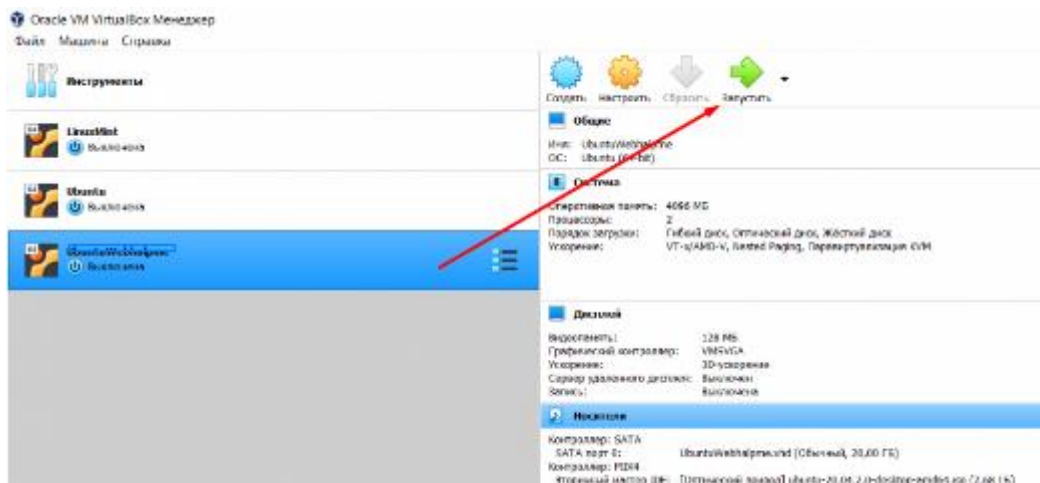


Рисунок 27 – Форма «Настройка ISO»

Поставьте везде галочки и нажмите «Продолжить». Рекомендуется выбрать «Обычная установка», также «Загрузить обновления во время установки...» и «Установить стороннее ПО» (рис. 28, 29).

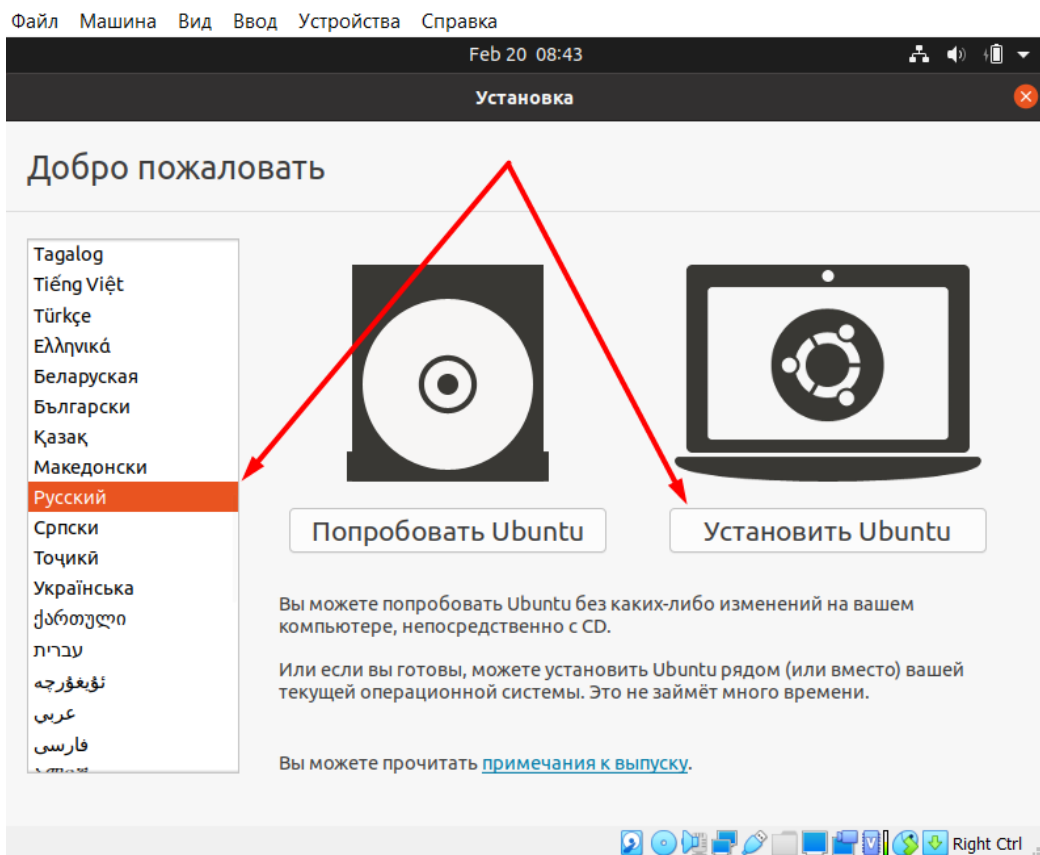


Рисунок 28 – Форма «Установка»

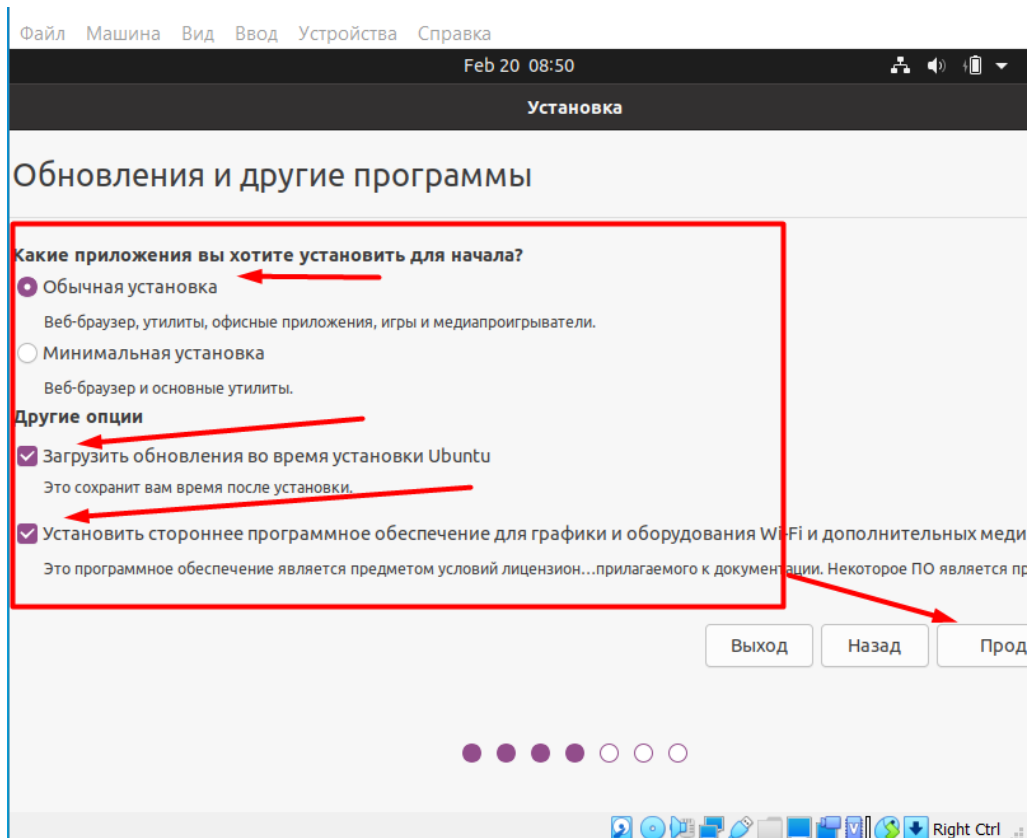


Рисунок 29 – Форма «Обновления и другие программы»

Следующим этап – «Тип установки» (рис. 30).

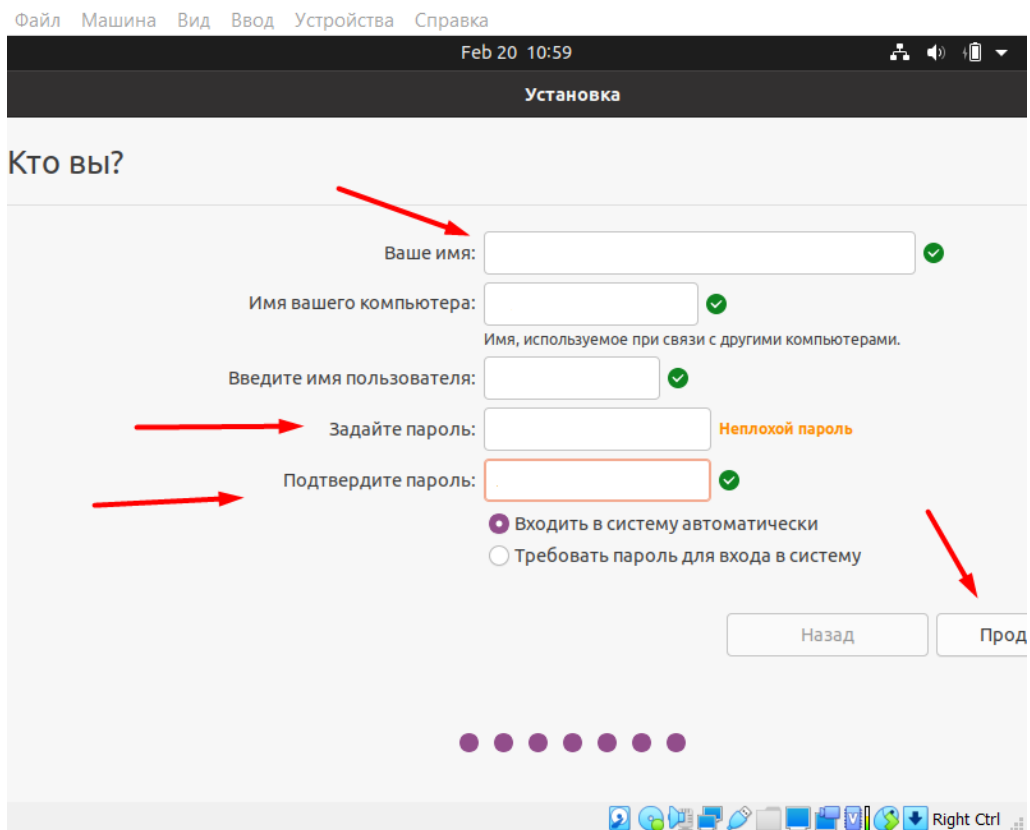


Рисунок 30 – Форма «Установка»

Необходимо выбрать (Стереть диск и установить Ubuntu), затем нажмите «Установить», после чего подтвердите свои действия нажав «Продолжить». Придумайте логин и пароль. Нажмите «Продолжить».

Пока идет копирование файлов, можете ознакомиться с Ubuntu в познавательном слайдшоу. По завершению копирования увидите «Установка Завершена», остается лишь выполнить «Перезагрузить».

Контрольные вопросы

Лабораторная работа №4

Тема: Работа с командной строкой в операционной системе Linux Ubuntu

Цель: Развитие навыков работы с командной строкой в операционной системе семейства Windows и умений по созданию командных файлов

Задание

1. Переключитесь на новую консоль, и войдите пользователем user, пароль user (ранее создайте пользователя с правами root).
2. Перейдите в каталог /dev (cd) и выведите на экран список содержащихся в нем файлов (ls), затем содержимое любого файла на экран (cat). При необходимости используйте команды (less и more)
3. Создайте пользователя, установите ему пароль, удалите пользователя.
4. Перейдите в папку /home и найдите в файле Lab1 строку с символом *
5. Найдите в файле Lab2 строку содержащую «???»
6. Содержимое файла Lab2 добавляем после последней строки в файл Lab1 (cat, перенаправление вывода)
7. Найти все появления слова "test" в файле schedule. Запустите команду grep в фоновом режиме и перенаправьте вывод в файл testfile. Затем выведите на экран содержимое testfile
8. Запустите команду grep (поиск во всех файлах в вашем текущем каталоге строки "word" перенаправив вывод в файл) в фоновом режиме, и покажите как она отображается в списке процессов
9. Завершите выполнение процесса grep принудительно.
10. Повторите задание 9 с использованием nohup

Linux – многозадачная, многопользовательская сетевая операционная система, поддерживающая стандарты открытых систем и протоколы сети Интернет. Все компоненты системы, включая исходные тексты, распростра-

няются с лицензией на свободное копирование и установку для неограниченного числа пользователей.

В создании системы принимают участие тысячи разработчиков по всему миру. Авторские права на каждую из программ сохраняются за ее создателями, сами программы свободно распространяются на основе Публичной лицензии GNU, предусматривающей свободное копирование и модификацию программ с обязательным указанием авторов первичных кодов и без права на коммерческое использование.

Преимущества Linux:

- легальное и бесплатное получение современной ОС;
 - высокое быстродействие;
 - надежная и устойчивая работа;
 - защита от вирусов;
 - предъявление очень скромных требований к ресурсам компьютера и, в то же время, очень эффективное использование возможностей, предоставляемых современными (в том числе многопроцессорными) системами;
 - эффективная реализация режима истинной многозадачности;
 - простое интегрирование компьютера в локальную сеть;
 - выполнение представленных в виде загрузочных модулей прикладных программ, написанных для других ОС;
- и пр.

Работа с консолью

Консолью называется совокупность основных устройств ввода информации в компьютер (клавиатура и мышь) и вывода информации (монитор). Linux работает с несколькими так называемыми виртуальными консолями, из которых в каждый момент времени только одна может быть связана с реальной (физической) консолью (то есть, является активной). В дальнейшем слово ``консоль" будет обозначать именно виртуальную консоль.

Консоли нумеруются целыми положительными числами. Их общее количество может изменяться в зависимости от настроек ОС и достигать нескольких десятков, хотя в стандартной настройке оно не превышает десяти. Несколько первых консолей – текстовые, далее идут графические (в стандартной настройке – одна).

При работе в графической консоли для того чтобы сделать активной другую консоль с некоторым номером *n*, требуется нажать клавиши **[Ctrl]+[Alt]+[Fn]**, то есть, например, **[Ctrl]+[Alt]+[F2]**, при переходе в консоль с номером 2.

Если в результате запуска корректно настроенной системы или при переходе из другой консоли активной становится текстовая консоль, следует ввести имя пользователя, которое вводится в позицию текстового курсора после приглашения `localhost login:` (здесь слово `localhost` означает имя компьютера в локальной сети по умолчанию; это имя может быть другим).

В следующей строке будет выведено приглашение `Password:`

При наборе пароля символы на экране не появляются, и курсор не перемещается.

При неверном вводе будет предложено повторить всю процедуру идентификации пользователя. В случае успешного прохождения идентификации появится приглашение командного процессора вида:

```
[user\_name@localhost dir\_name]$
```

(`user_name` – имя пользователя, а `dir_name` – название текущего каталога).

Дальнейшая работа состоит в запуске различных программ в режиме командной строки.

Общая структура файловой системы

Файловая система Linux основана на модели иерархического дерева каталогов. Однако в Linux отсутствует понятие логического устройства (диска), все каталоги являются подкаталогами единого дерева и начинаются с так

называемого корневого каталога. Корневой каталог системы обозначается символом /, подкаталог корневого каталога с именем каталог1 обозначается /каталог1, подкаталог этого каталога /каталог1/каталог2, а файл, находящийся в каталоге /каталог1 обозначается /каталог1/файл1 (никакой разницы в обозначении файлов и каталогов не существует).

В именах файлов и каталогов могут встречаться практически любые символы (причем прописные и строчные буквы различаются), однако далеко не все программы могут работать с именами, в которых используется, например, символ звездочка (*). Необходимо проявлять особую осторожность и при работе с файлами, содержащими в именах символы с кодами, превышающими 127 (например, буквы национальных алфавитов).

Имена физических устройств компьютера выглядят как имена файлов в подкаталоге первого уровня /dev (и действительно являются файлами особого вида). Разделы жесткого диска с интерфейсом IDE (EIDE) имеют имена вида /dev/hdXY, где X это одна из букв a, b, c, d, обозначающие соответственно с 1 по 4 физический диск (от Primary Master до Secondary Slave), а Y число, обозначающее номер раздела на диске (разделы нумеруются в том порядке, в котором они перечислены в таблице разделов диска). Например, единственный раздел второго (Slave) диска, присоединенного к первичному (Primary) контроллеру, обозначается /dev/hdb1.

Присоединение других файловых систем (монтирование устройств)

Для того чтобы сделать доступными файловые системы, размещенные на физических устройствах, требуется совместить корневые каталоги этих устройств с некоторыми подкаталогами файловой системы. Эта операция называется монтированием устройства. Жесткие диски обычно монтируются при запуске системы (например, основной раздел установки Linux, монтируется в корневой каталог /), однако их можно монтировать и отдельно.

Другие устройства, например, дисководы для гибких дисков А: (устройство /dev/fd0) и В: (/dev/fd1) и дисковод CD-ROM (/dev/cdrom), монтируются по мере необходимости.

По окончании работы с устройствами они размонтируются, и размещенные на них файлы становятся недоступными. Для жестких дисков это действие обычно выполняется автоматически при отключении системы. Иначе обстоит дело с гибкими дисками и CD-ROM: соответствующие устройства необходимо размонтировать перед тем как удалять из компьютера носители.

Нельзя удалить компакт-диск из неразмонтированного привода (устройство не реагирует на кнопку извлечения).

Команда монтирования устройства имеет вид:

```
mount -t тип_файловой_системы устройство каталог
```

Параметр устройство задает физическое устройство в вышеописанном формате, параметр каталог точку монтирования (каталог общего дерева каталогов, с которым будет совмещен корневой каталог устройства, каталог должен существовать и быть пуст), параметр тип_файловой_системы ключевое слово, обозначающее стандарт, в котором записываются на устройстве файлы и каталоги. Чаще всего могут встретиться следующие типы файловых систем:

ext2

файловая система, используемая в разделах жесткого диска GNU/Linux;

iso9660

файловая система, используемая на большинстве CD-ROM;

vfat

файловая система, используемая (с вариациями) в MS-DOS и Windows 9x для жестких дисков и дискет (включает в себя системы, в терминологии Windows называемые FAT и FAT32).

Таким образом, чтобы смонтировать привод CD-ROM в каталог /MyCD, требуется ввести команду

```
mount -t iso9660 /dev/cdrom /MyCD
```

Для двух наиболее стандартных ситуаций можно применять упрощенные форматы команд монтирования.

Для того чтобы смонтировать дисковод A: на каталог /mnt/floppy с автоматическим определением типа файловой системы дискете, введите

```
mount /dev/fd0
```

Для того чтобы смонтировать дисковод CD-ROM на каталог /mnt/cdrom с автоматическим определением типа файловой системы на диске, введите

```
mount /dev/cdrom
```

Для того чтобы размонтировать устройство, введите команду

```
umount устройство
```

например, `umount /dev/cdrom`

Домашний каталог

Каждый пользователь системы имеет свой личный каталог, который называется начальным каталогом или домашним. При входе в систему после ввода имени и пароля вы окажетесь в вашем личном каталоге. Имя начального каталога пользователя хранится в системной переменной ``\$HOME''.

Каталог может содержать и другие каталоги (подкаталоги). Поэтому естественным способом представления организации каталогов и файлов является дерево каталогов. В процессе обхода этого дерева, начиная от корня, можно найти любой нужный файл.

Каталог является специальным файлом, который содержит имена файлов, размещенных в этом каталоге, а также имена подкаталогов и ссылки на них. По отношению к подкаталогам, текущий каталог является родительским (parent directory). Для обозначения каталогов используются следующие системные имена:

. (точка) - текущий каталог

.. (две точки) - родительский каталог, расположенный на один уровень ближе к корню.

Структура каталогов Linux

Только что установленная система Linux имеет дерево каталогов следующей структуры:

/

корневой каталог.

/bin

основные системные программы.

/boot

загрузочные файлы ядра ОС.

/dev

описания устройств компьютера.

/etc

конфигурационные файлы системы и подкаталоги с конфигурационными файлами прикладных программ.

/home

подкаталоги (домашние каталоги) пользователей.

/lib

динамические библиотеки.

/lost+found

информация об удаленных файлах, при некоторых условиях помогающая восстановить данные.

/mnt

подкаталоги стандартные точки монтирования сменных устройств, таких как дисковод CD-ROM.

/root

домашний каталог суперпользователя.

/sbin

системные программы.

/usr

прикладные программы и библиотеки.

/var

рабочие каталоги программ.

Следует помещать свои документы и другие файлы только в свой домашний каталог. В нем можно создавать любые нужные подкаталоги, руководствуясь соображениями удобства представления информации. Не следует вносить изменения в другие каталоги, за исключением случаев установки новых прикладных программ (в этом случае надо следовать инструкции к дистрибутиву устанавливаемой программы) или редактирования конфигурационных файлов.

Многопользовательские возможности

Многопользовательские возможности системы Linux имеют принципиальное значение: любые действия пользователя возможны только после идентификации пользователя с какой-либо из регистрационных записей. Linux предоставляет пользователю возможности, определяемые его правами на работу с файлами и каталогами. Считается, что каждый файл принадлежит определенному пользователю и определенной группе пользователей. С каждым файлом также связана таблица прав, указывающая, какие действия (чтение, запись, исполнение программы, открытие каталога и некоторые другие) может предпринимать владелец файла, член группы, владеющей файлом, и произвольный пользователь.

Кроме регистрационных записей обычных пользователей существует также регистрационная запись суперпользователя с зарезервированным именем root. Суперпользователь может выполнять любые допустимые действия с любым файлом, независимо от того, кто является его владельцем и какова таблица прав этого файла. Кроме того, суперпользователь может изменять информацию о владельце любого файла и его таблицу прав. А также суперпользователь обладает значительно более широкими правами, не связанными с файловыми операциями.

Создание групп, состоящих из пользователей, совместно работающих над определенными проектами, и присвоение каталогам, содержащим файлы этих проектов, соответствующего атрибута группы-владельца, в сочетании с настройкой таблицы прав, определяющих, какие именно операции могут выполнять члены этой группы, дает мощный инструмент для совместной работы над защищенными от постороннего вмешательства данными.

Добавление пользователя

Для того чтобы добавить регистрационную запись обычного пользователя, требуется сделать следующее.

Перейдите в текстовую консоль или режим эмуляции терминала и войдите в систему как суперпользователь. Введите в командной строке команду

```
adduser имя_пользователя
```

где имя_пользователя должно состоять из латинских букв и цифр и начинаться с буквы. В системе не должно существовать регистрационной записи с таким же именем. В случае успешного завершения операции создания новой регистрационной записи никаких сообщений не выдается. Создание регистрационной записи происходит в режиме, когда вход пользователя невозможен, поэтому для вновь созданного пользователя необходимо установить пароль.

Команда `adduser` создает регистрационную запись для нового пользователя (для чего вносит изменения в ряд конфигурационных файлов системы) и создает для него так называемый домашний каталог

```
/home/имя_пользователя
```

содержащий некоторые необходимые файлы и подкаталоги. Этот каталог и файлы в нем принадлежат пользователю, и он имеет полный набор прав для работы с этими файлами и подкаталогами, а также для всех вновь создаваемых им в этом каталоге объектов.

Установка или смена пароля

Для введения пароля нового пользователя или изменения пароля существующего введите в командной строке команду

```
passwd имя_пользователя
```

В ответ на приглашение

```
New UNIX password:
```

введите пароль (допускаются любые символы, прописные и строчные буквы различаются). На экране не отображаются никакие символы, и курсор по мере набора символов не перемещается.

Если введенный пароль слишком простой (например, короче шести символов), об этом будет выдано предупреждение, начинающееся со слов BAD PASSWORD (далее следует описание недостатков пароля).

После ввода пароля будет предложено подтвердить введенный пароль. В ответ на приглашение

```
Retype new UNIX password:
```

введите пароль еще раз.

В случае успеха (если дважды был введен одинаковый пароль) выдается сообщение

```
passwd: all authentication tokens updated successfully
```

и программа завершает работу. Если пароли, введенные в первом и во втором случае, не совпадают, выдается сообщение

```
Sorry, passwords do not match
```

и Вам вновь предлагается ввести пароль.

Удаление пользователя

Иногда необходимо удалить регистрационную запись пользователя (например, зарегистрированного по ошибке). Для того чтобы удалить регистрационную запись пользователя имя_пользователя, необходимо работать в текстовой консоли с правами суперпользователя. Введите в командной строке команду

```
userdel имя_пользователя
```


Для того чтобы удалить регистрационную запись, а также принадлежащие пользователю файлы, введите в командной строке команду

```
userdel -r имя_пользователя
```

Эта команда удаляет каталог /home/имя_пользователя и все его содержимое. Файлы данного пользователя, расположенные в других местах, не удаляются автоматически, их необходимо находить и удалять вручную.

Удаление файлов, принадлежавших ранее удаленному пользователю, а также любой другой доступ к ним (например, просмотр) возможен только с правами суперпользователя.

Удаление пользователя невозможно, если он в данный момент зарегистрирован в системе или работает какой-либо процесс, запущенный от его имени.

Командный процессор

При работе пользователя в текстовой консоли или терминале первичный диалог осуществляет системная программа, называемая командным процессором. В простейшем случае эта программа выдает приглашение и ожидает ввода команды. Команда представляет собой имя исполняемого файла (двоичного или текстового, так называемого скрипта, написанного на одном из специальных командных языков) или имя внутренней команды самого процессора. По окончании работы эти программы, как правило, возвращают управление командному процессору, однако могут также возвращать управление, не завершая работы (при этом командный процессор выдает приглашение, а ранее запущенная программа продолжает работу в фоновом режиме).

Linux может использовать разные командные процессоры, в том числе несколько одновременно, однако Вам, вероятнее всего, встретится наиболее распространенный из них Bash. Shell - интерпретатор командного языка

Запуск команды в фоновом режиме (символ &)

Некоторые команды shell занимают много времени при выполнении. Эти команды можно запустить в фоновом режиме с использованием &, освобождая тем самым терминал для других задач. Общий формат для запуска команд в фоновом режиме следующий:

```
command &
```

Примечание. Интерактивные команды shell (например, read) нельзя запускать в фоновом режиме.

Последовательное выполнение команд (символ ;)

В одной командной строке вы можете указать несколько команд. Эти команды должны быть разделены символом ; (точка с запятой) или символом & (амперсанд):

```
command1; command2; command3
```

Отмена специального значения (метасимвол ``слэш'')

Символ \`(`слэш`)` позволяет вам отменить специальное значение следующего за ним символа. Например, у вас есть файл trail, который содержит следующий текст:

```
The all * game  
was held in Summit.
```

Чтобы найти символ звездочка (*) в файле, воспользуйтесь командой grep:

```
$ grep \* trail  
The all * game  
$
```

команда grep найдет символ * в тексте и отобразит строку, в которой она появилась. Без символа \, символ звездочка будет интерпретироваться shell как метасимвол.

Отмена специального значения (метасимвол кавычки)

Отменить специальное значение символа вы также можете с помощью метасимвола кавычки. Одиночные кавычки ('...') отменяют специальное значение всех символов за исключением самих одиночных кавычек. Двойные кавычки ("...") отменяют специальное значение всех символов, за исключением символов двойные кавычки, \$ и ` (слабое ударение). Использование кавычек удобно для цифровых специальных символов.

Например, файл trail содержит строку:

```
He really wondered why? Why???
```

Чтобы найти строку, содержащую три вопросительных знака, воспользуйтесь командой `grep`:

```
$ grep '???' trail
```

```
He really wondered why? Why???
```

```
$
```

Перенаправление ввода и вывода

В системе Linux некоторые команды ожидают ввод только с клавиатуры (стандартный ввод) и большинство команд отображают свой вывод на экране терминала (стандартный вывод). Однако система Linux позволяет вам перенаправлять ввод и вывод в файлы и программы, т.е. вы можете сказать shell:

взять ввод из файла, а не с клавиатуры;

послать вывод в файл, а не на терминал;

использовать программу как исходные данные для другой программы.

Чтобы перенаправить ввод, укажите в командной строке после знака "меньше чем" (<) имя файла:

```
command < имя\_файла
```

Чтобы перенаправить вывод, укажите в командной строке после знака "больше чем" (>) имя файла:

```
command > имя\_файла
```

Примечание. Если вы перенаправите вывод в уже существующий файл, то вывод вашей команды заменит содержимое существующего файла.

Перед тем, как перенаправить вывод команды в конкретный файл убедитесь, что этот файл не существует. shell не предупреждает, что выполняет перезапись существующего файла.

Добавить вывод в существующий файл

Чтобы добавить вывод в существующий файл и не разрушить его, можете воспользоваться символом »:

```
command >> имя_файла
```

Команда cat печатает содержимое файлов, имена которых являются ее аргументами, в стандартный вывод.

Комбинирование фонового режима и перенаправления вывода

Когда команда запущена в фоновом режиме, то вывод ее печатается на экране терминала. И если используете терминал в то же время для выполнения других задач, то вывод фоновой задачи будет прерывать вашу работу. Однако, если перенаправить вывод в файл, то вы сможете спокойно работать.

Получить состояние запущенного процесса

Команда ps дает состояние всех процессов, запущенных на данный момент. Например, вы можете использовать команду ps, чтобы просмотреть состояние всех процессов, которые запущены в фоновом режиме, применив символ &.

В следующем подпункте обсуждается вопрос, как применить номер PID (идентификатор процесса), чтобы остановить выполнение команды. PID является уникальным номером, который система Linux назначает каждому активному процессу.

Обратите внимание, что система распечатала номер PID для команды gper так же, как и для всех других запущенных процессов: для самой команды ps и команды bash, которая была запущена во время вашей регистрации.

Можно приостановить и вновь запустить программу, если в системе предусмотрена функция управления заданиями. Команда jobs выдает список текущих фоновых процессов, запущенных или приостановленных. Команда jobs дополнительно к PID распечатывает идентификатор задания (JID) и имя задания. Чтобы вновь запустить приостановленное задание, либо возобновить фоновый процесс в оперативном режиме, вам необходимо знать JID. JID распечатывается на экране каждый раз, когда вы вводите команду запуска или останова процесса. Если введете:

```
jobs
```

то на экране появится следующая информация:

```
[JID] - Stopped (signal) <имя задания>
```

или

```
[JID] + Running <имя задания>
```

Завершение активных процессов

Команда kill завершает активные процессы в фоновом режиме и команда stop приостанавливает временно процессы. Общий формат этих команд:

```
kill PID
```

или

```
stop JID
```

Обратите внимание, что вы не можете завершать фоновые задания нажатием клавиш BREAK или DEL.

После того как система выдаст ответ на запрос, на экране появится подсказка \$, означающая, что процесс завершен. Если система не найдет указанный PID, то появится сообщение об ошибке:

No such process

Чтобы приостановить оперативный процесс (если активна функция управления заданиями), введите `^Z`. На экране появится следующее сообщение:

```
<JID> Stopped(user) <имя задания>
```

Запуск остановленного задания

Если функция управления заданиями активна, то вы можете вновь запустить приостановленный процесс. Чтобы вновь запустить процесс, остановленный командой `stop`, сначала должны определить JID с помощью команды `jobs`. Затем можете использовать JID со следующими командами:

```
fg <JID>
```

- возобновляет приостановленное задание или переводит задание из фонового режима в оперативный;

```
bg <JID>
```

- вновь запускает приостановленное задание в фоновом режиме.

Если хотите, чтобы после вашего выхода из системы процесс в фоновом режиме продолжал выполняться, то вам необходимо использовать команду `nohup`. Команда `nohup` имеет следующий формат:

```
nohup command &
```

Лабораторная работа №5

Тема: Работа с образом операционной системы Android

Цель: Развитие навыков инсталляции и настройки параметров операционной системы Android

Задание

Выполнить настройку аппаратной части для Android OS

Осуществить запуск Android OS на виртуальной машине

Настроить работу с Android в VirtualBox

Технология выполнения работы

Для выполнения лабораторной работы этого нам понадобится:

1. Виртуальная машина VirtualBox.
2. Виртуальный ISO образ Android x86.
3. Свободное место на диске. Файл виртуальной машины потребует 8 ГБ свободного места на диске.

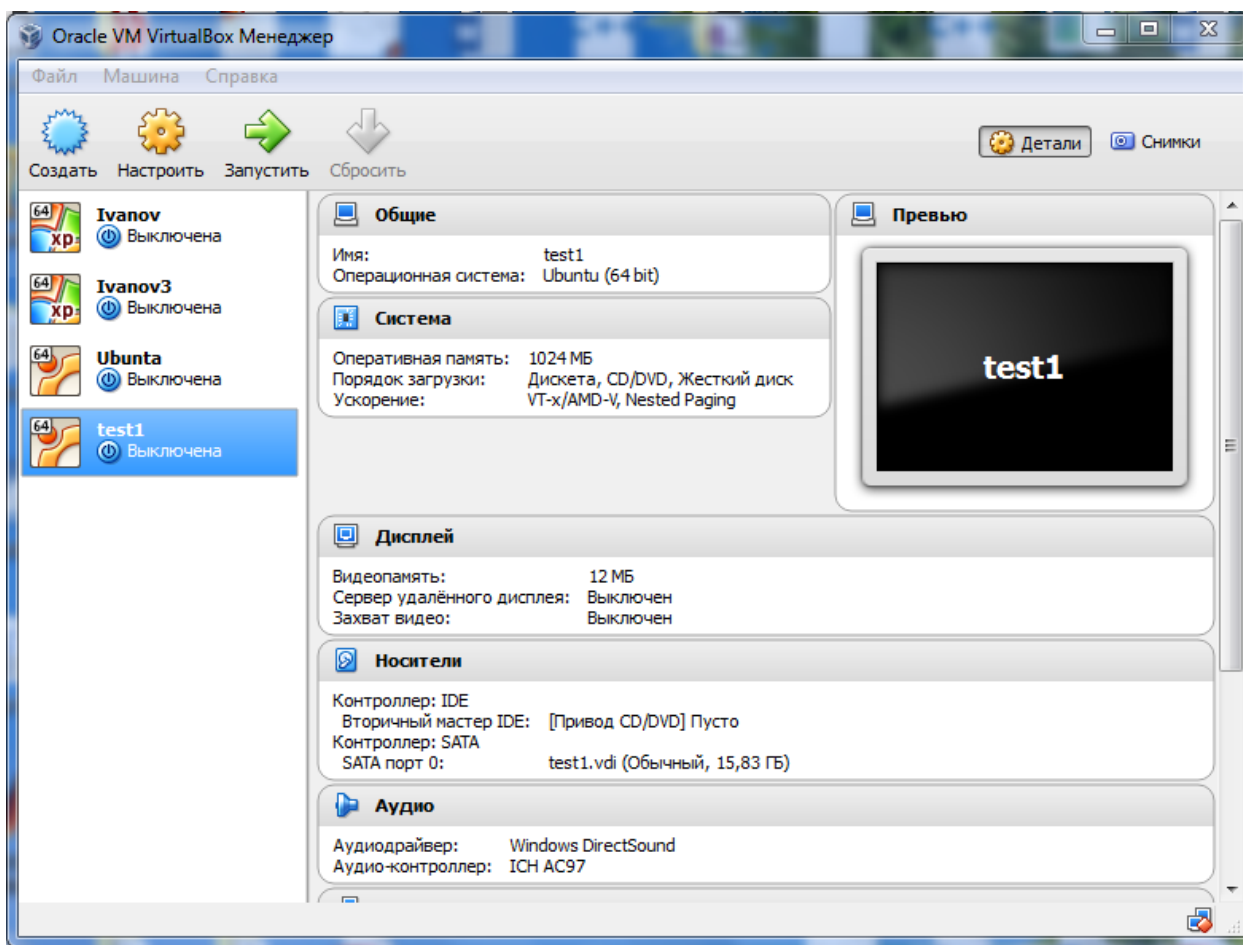
Прежде чем приступать к работе, рекомендуется убедиться, что параметры виртуализации Intel VT-x или AMD-V активированы в BIOS ПК. В противном случае получите ошибку при запуске виртуальной машины.

Настройка аппаратной части для Android OS

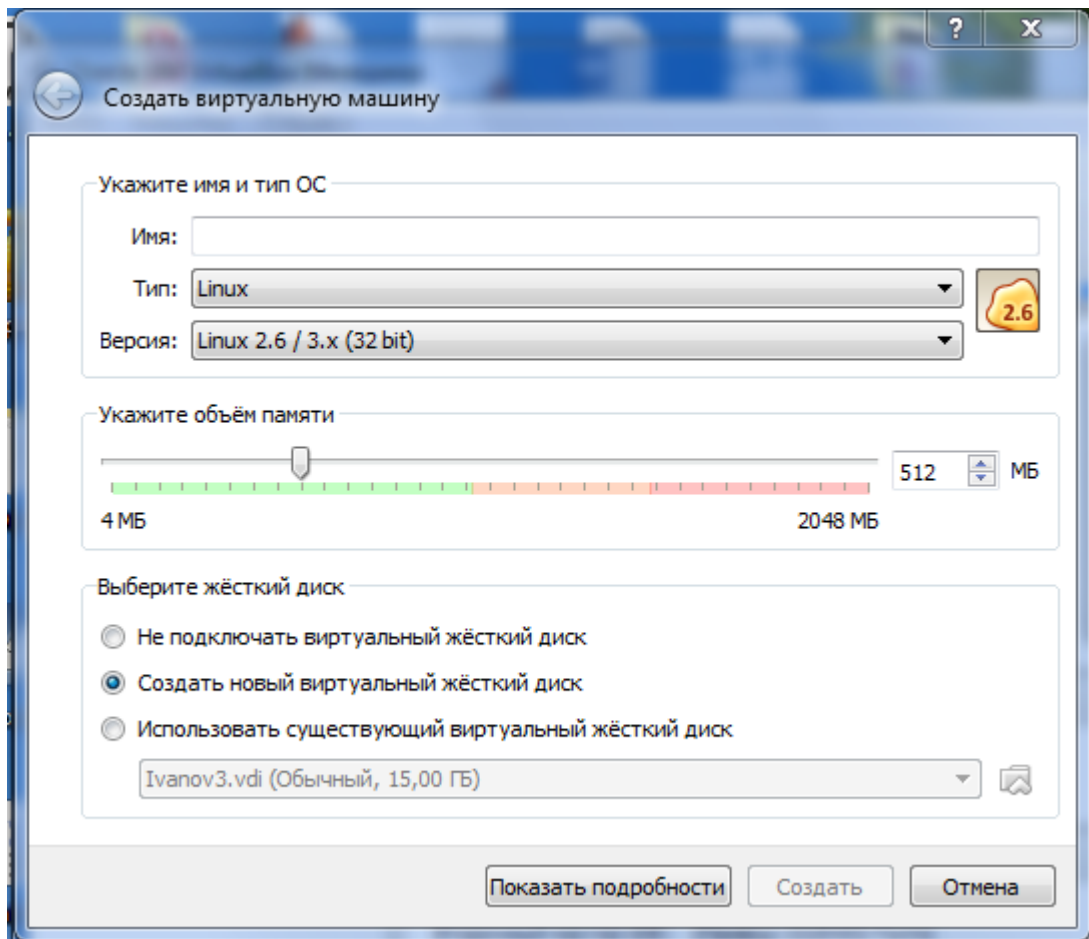
Перед установкой Android OS на виртуальную машину необходимо, создать аппаратную конфигурацию этой самой виртуальной машины. Выполните следующие пункты лабораторной работы.

1. Запустите программу Virtual Box.

2. Нажмите кнопку *Создать* расположенную на главной панели окна программы.

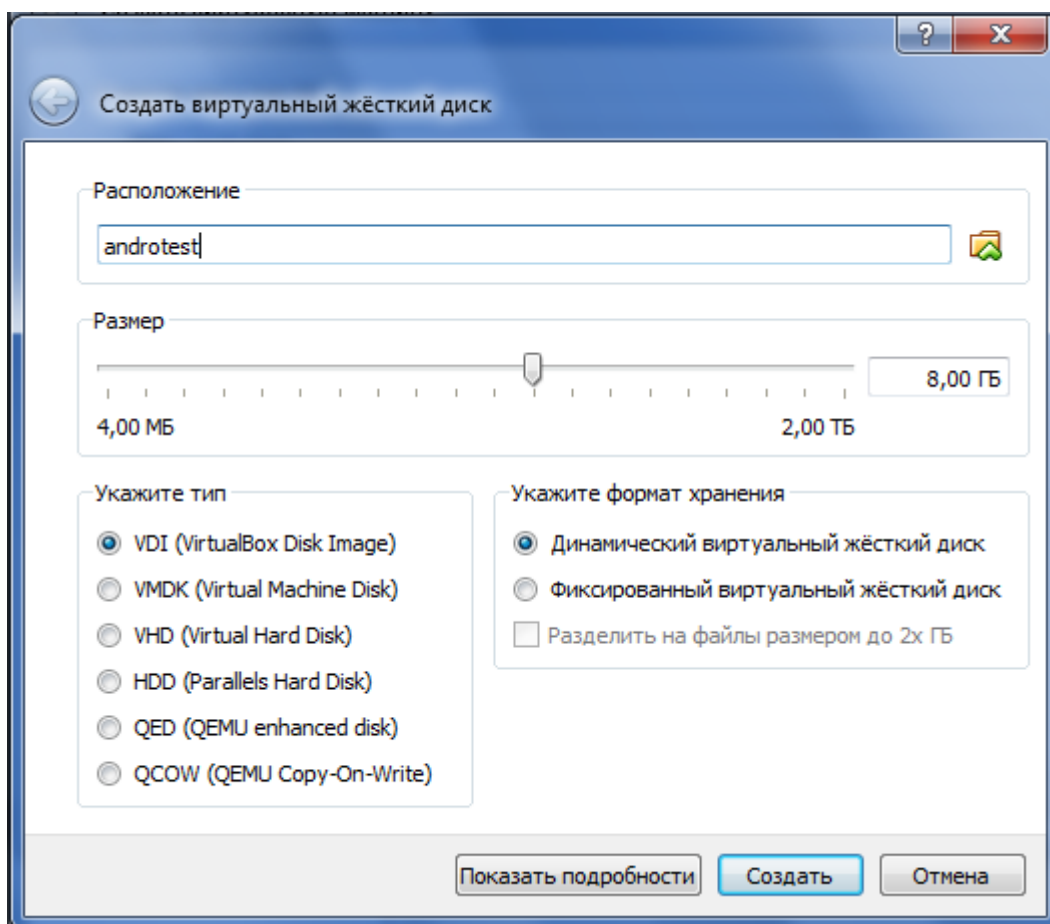


3. Далее укажите название виртуальной машины, тип и версию ОС. Тип системы укажите Linux, версию Linux 2.6 / 3.x / (32 бита). Нажмите Next для продолжения.



4. На следующем шаге укажите рекомендуемый объем памяти 1024 МБ и нажмите Next.

5. На следующем шаге необходимо создать новый виртуальный жесткий диск. Нажмите кнопку Создать для продолжения.



6. На следующем шаге необходимо выбрать тип диска. Укажите VDI (VirtualBox Disk Image) и нажмите Next для продолжения.

7. На следующем этапе необходимо указать вариант использования жесткого диска. Можно либо зарезервировать все место для диска сразу, либо разрешить увеличивать размер диска по мере использования. Выберите Фиксированный виртуальный жесткий диск для оптимизации скорости работы с диском и нажмите Next.

8. На следующем шаге необходимо указать имя файла и его размер, нажмите Создать для продолжения.

9. Дождитесь создание виртуальной машины.

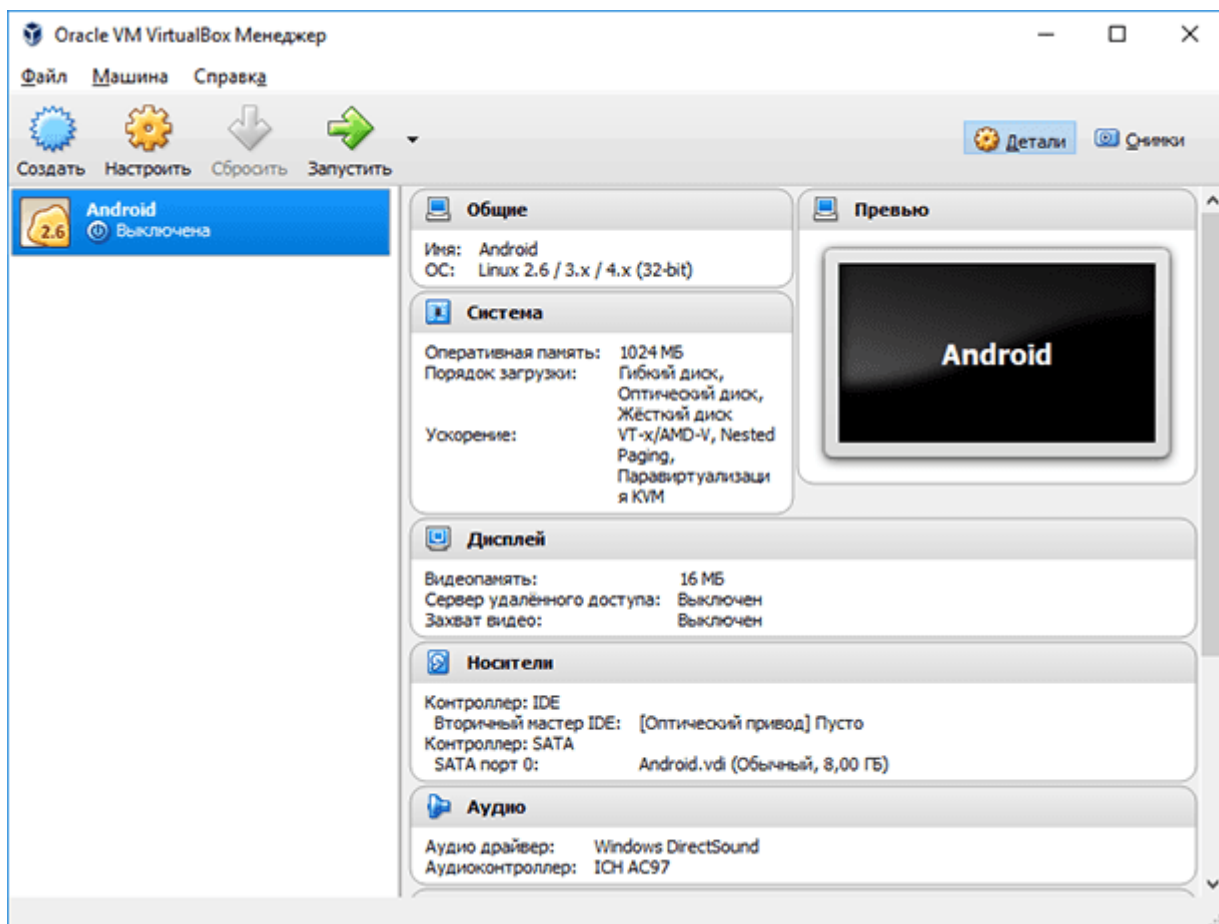
На этом настройка аппаратной части виртуальной машины закончена

Установка и запуск Android OS на виртуальной машине.

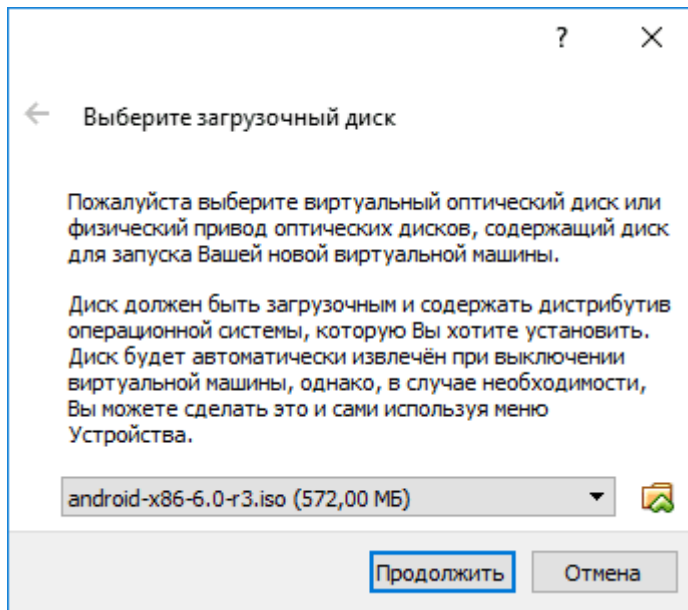
После выбора нужных “физических” параметров, и настройки аппаратной части, можно приступить к установке нужной вам ОС.

Выполните следующие шаги:

1. После настройки и создания виртуальной машины нажмите кнопку Запустить в главном меню.

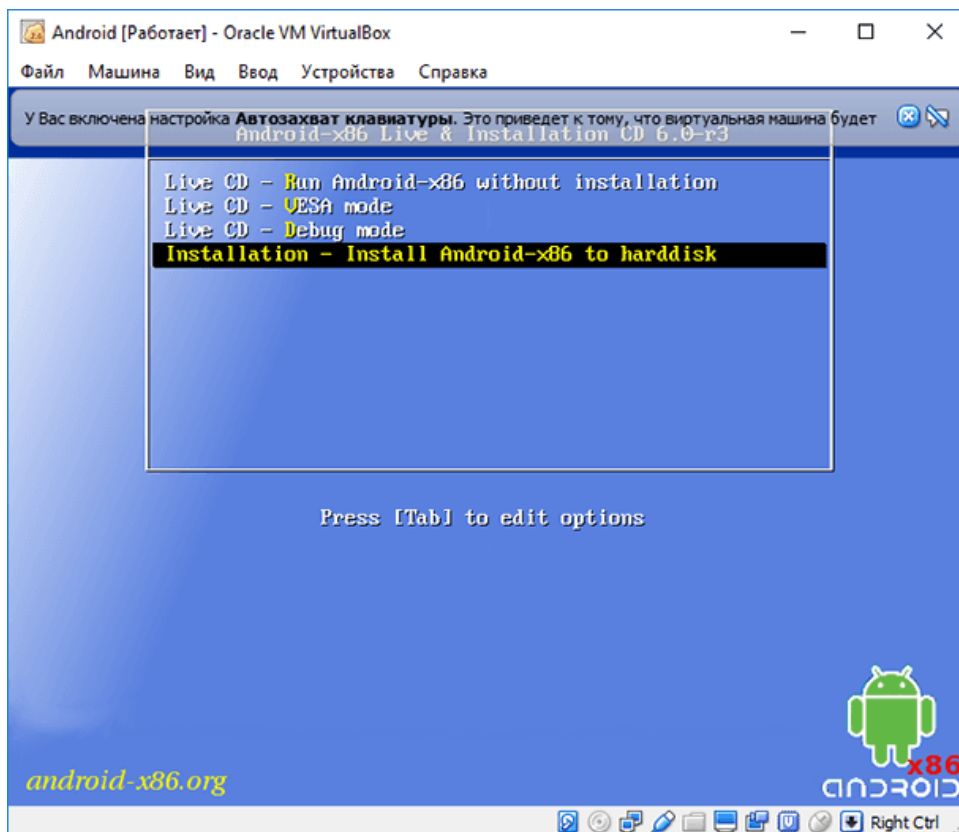


2. Когда машина начнет загрузку, укажите путь к ISO образу с Android и нажмите кнопку Продолжить. Если вы закрыли мастер, вы берите виртуальную машину в списке слева, затем в главном меню кликните на Устройства – Оптические диски – Выбрать образ диска. Затем выполните перезагрузку машины с помощью Машина – Перезагрузить.

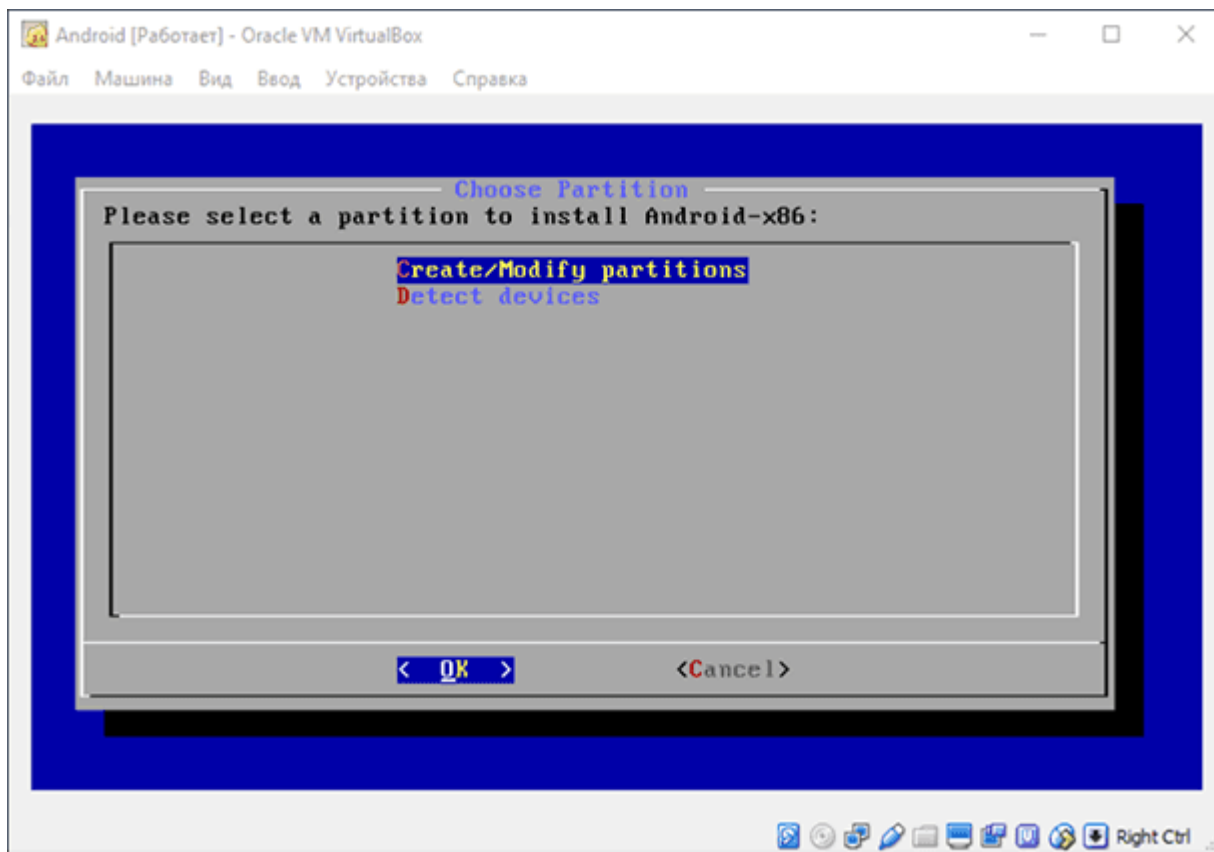


Важно: Когда вы кликните мышкой внутри окна VirtualBox, оно автоматически перехватит мышку и клавиатуру. Что бы высвободить мышь и клавиатуру нажмите правую клавишу Ctrl на клавиатуре.

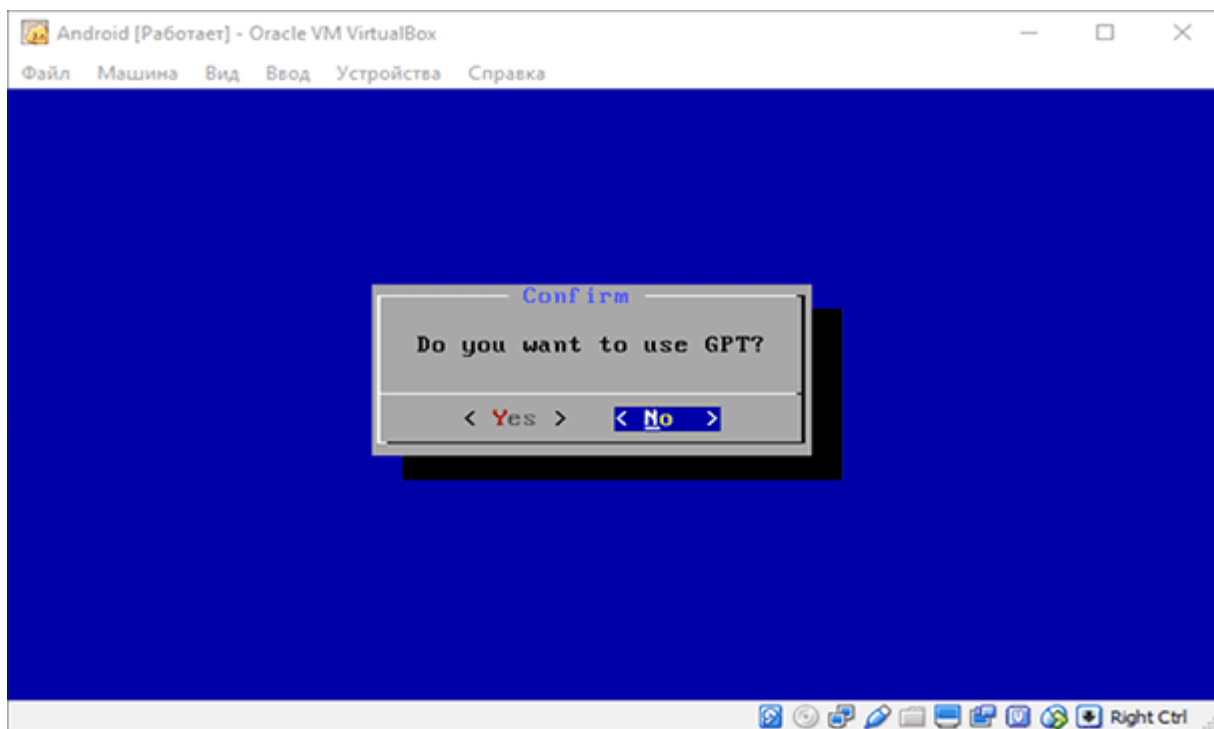
3. После загрузки виртуальной машины используйте клавиатуру для выбора пункта Installation – Install Android-x86 to harddisk и нажмите Enter для продолжения.



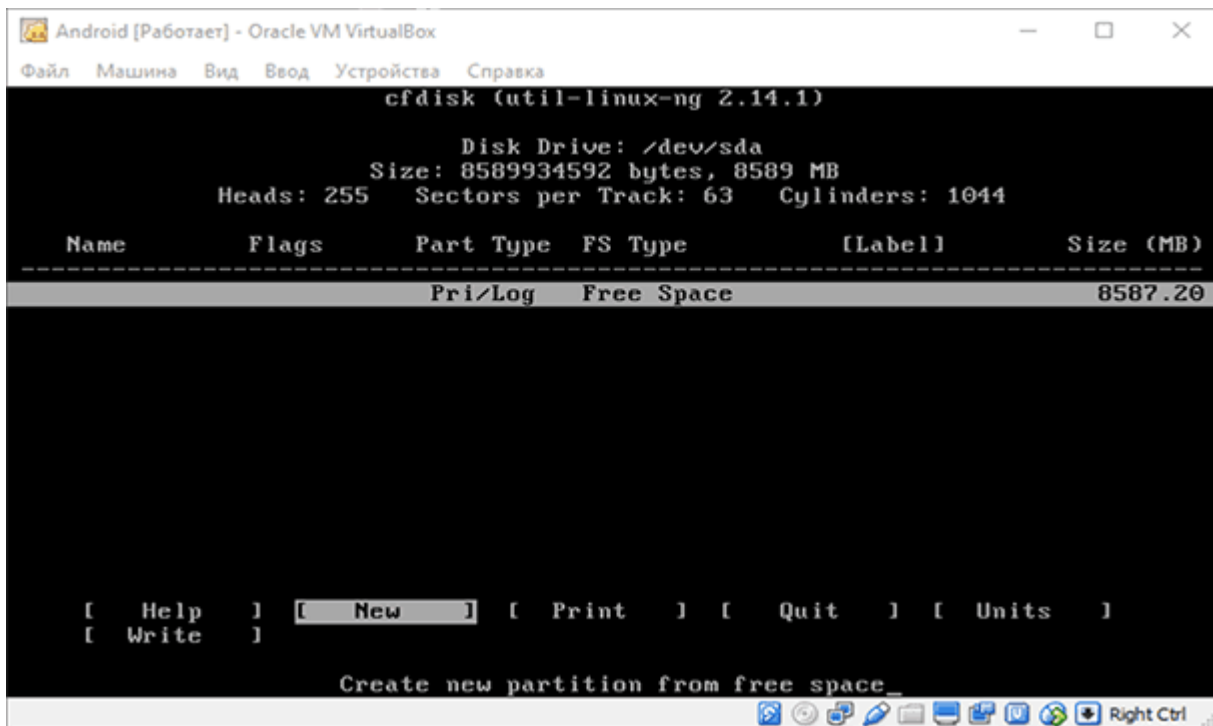
4. На следующем шаге выберите Create/Modify partitions, подтвердите выбор Enter.



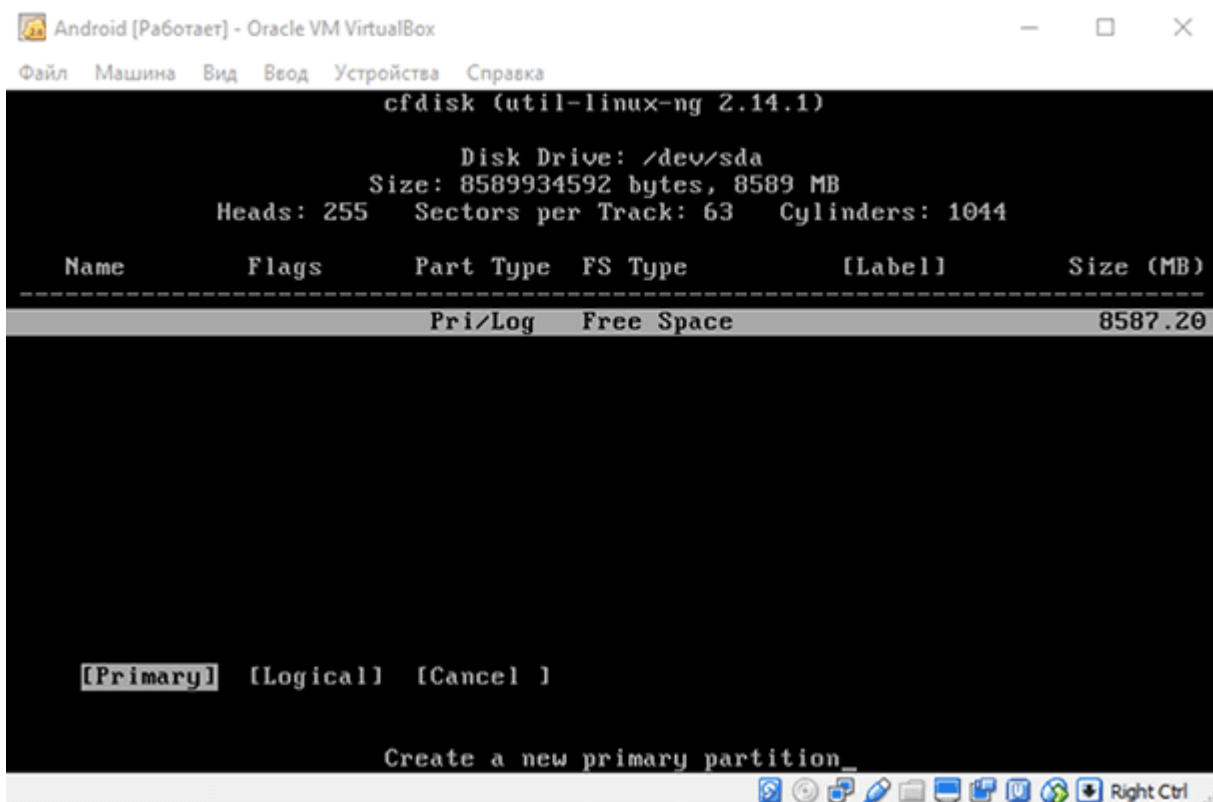
5. На следующем шаге выберите не использовать GPT.



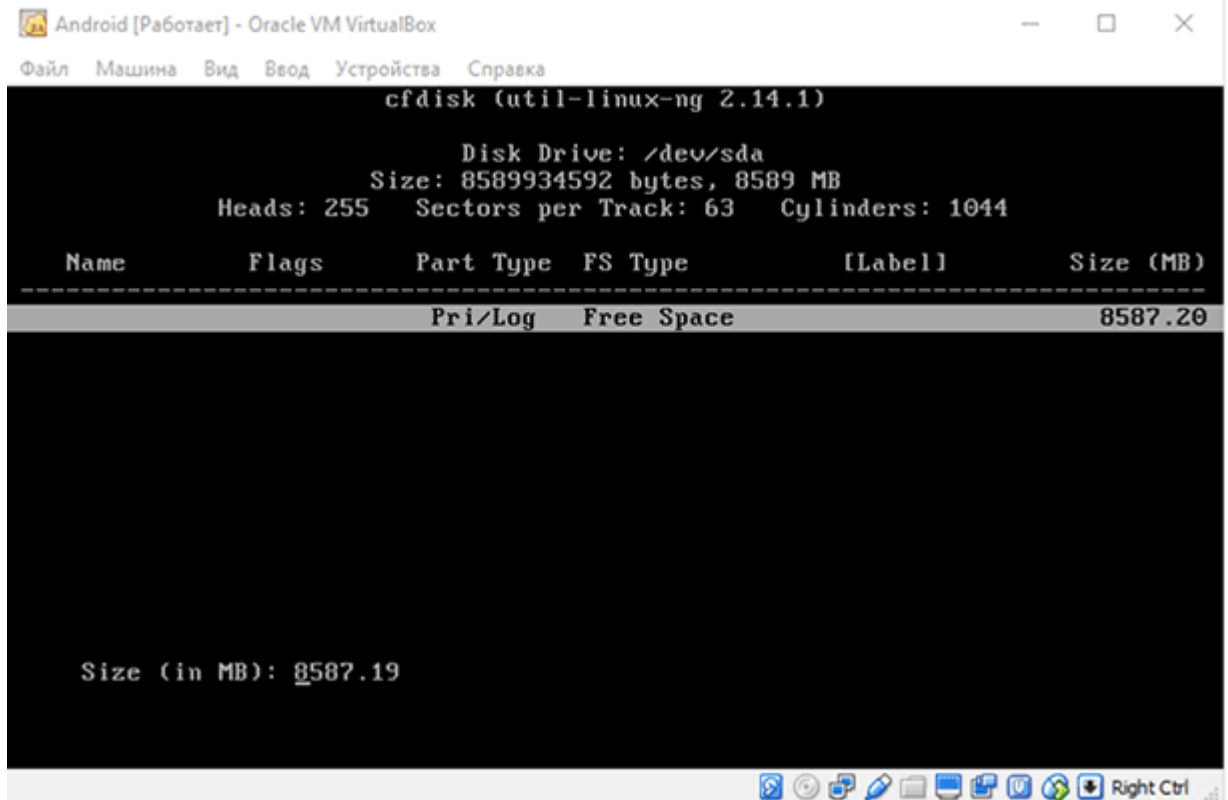
6. На следующем шаге откроется утилита для работы с разделами жесткого диска. С помощью клавиатуры выберите New и Enter для продолжения.



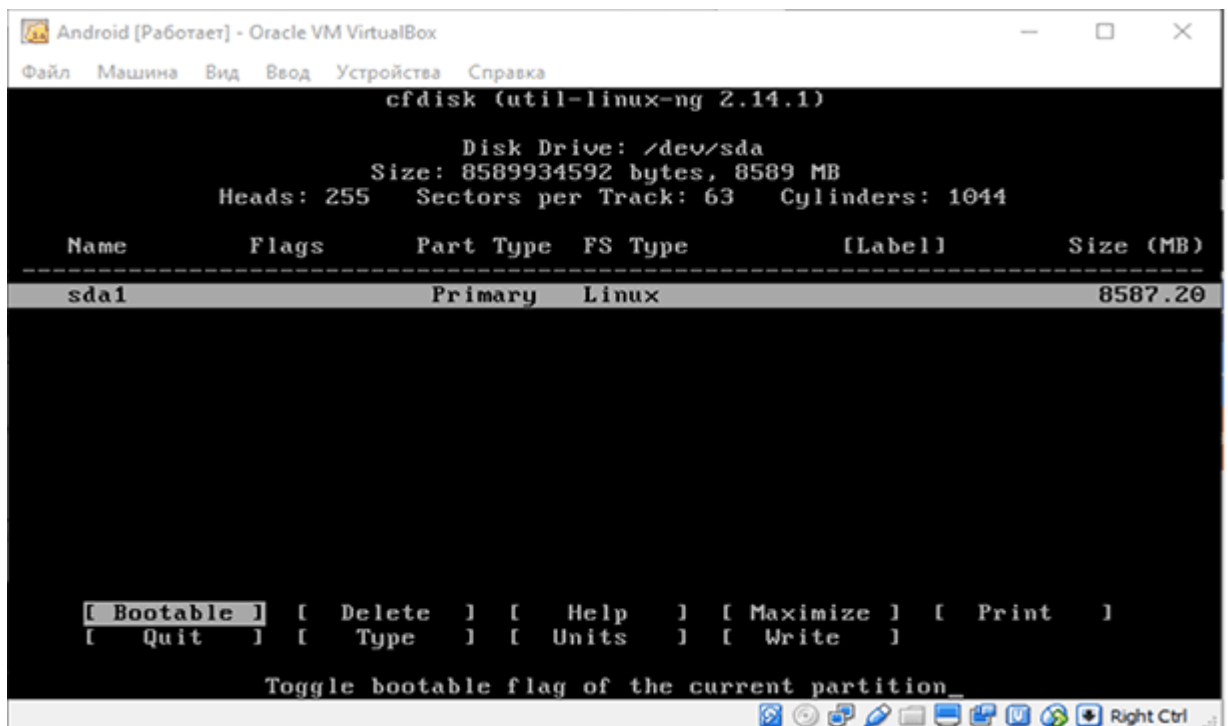
7. На следующем шаге укажите тип раздела – Primary и нажмите Enter для продолжения.



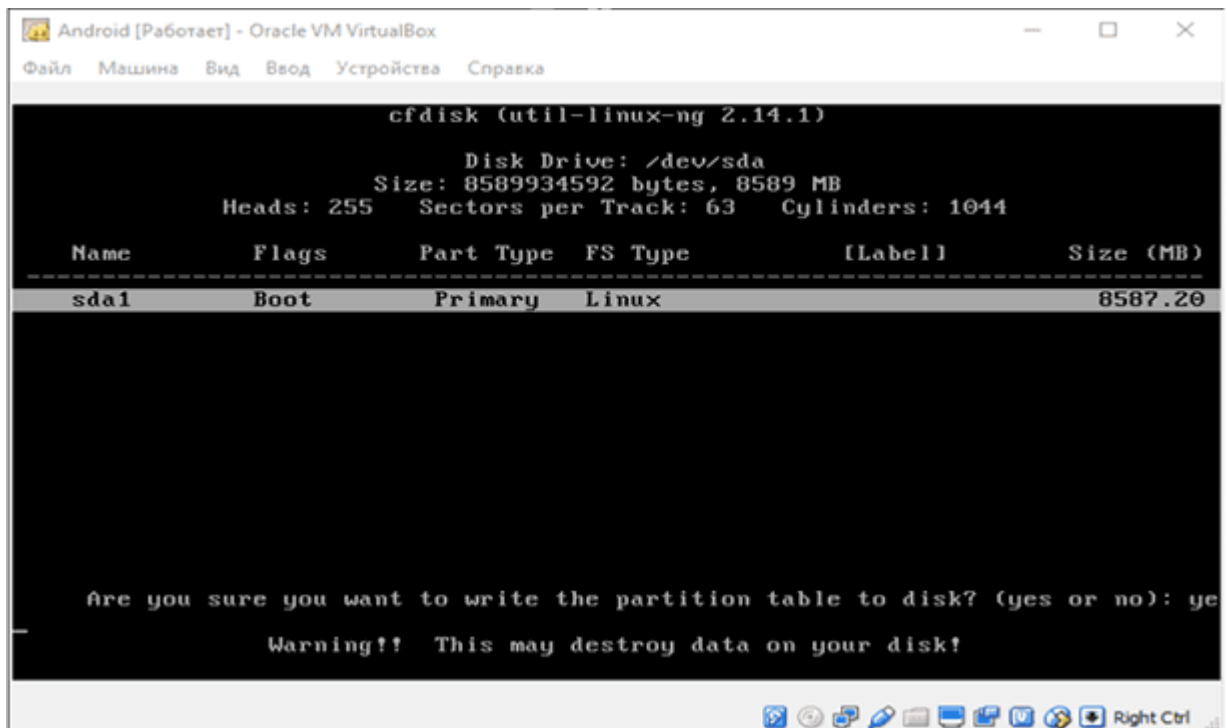
8. Подтвердите размер диска равный 8Gb с помощью клавиши Enter.



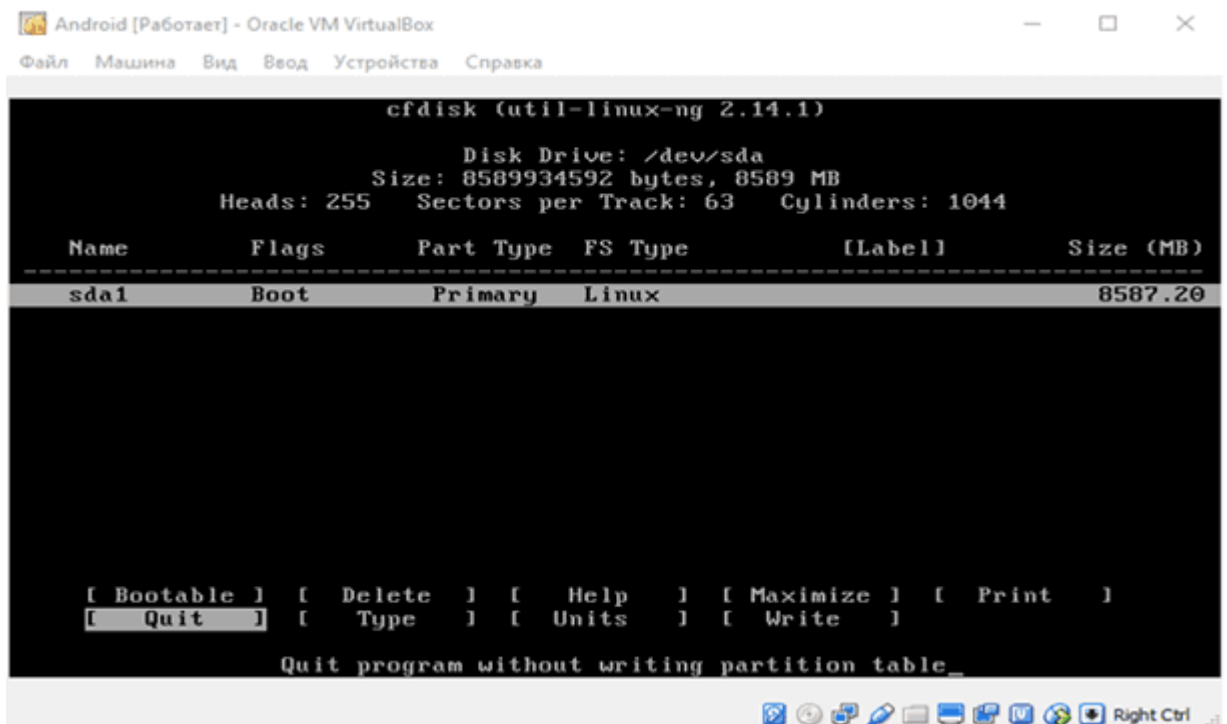
9. На следующем шаге назначьте новому диску атрибут Bootable. Используйте стрелки на клавиатуре для выбора атрибута и Enter для назначения.



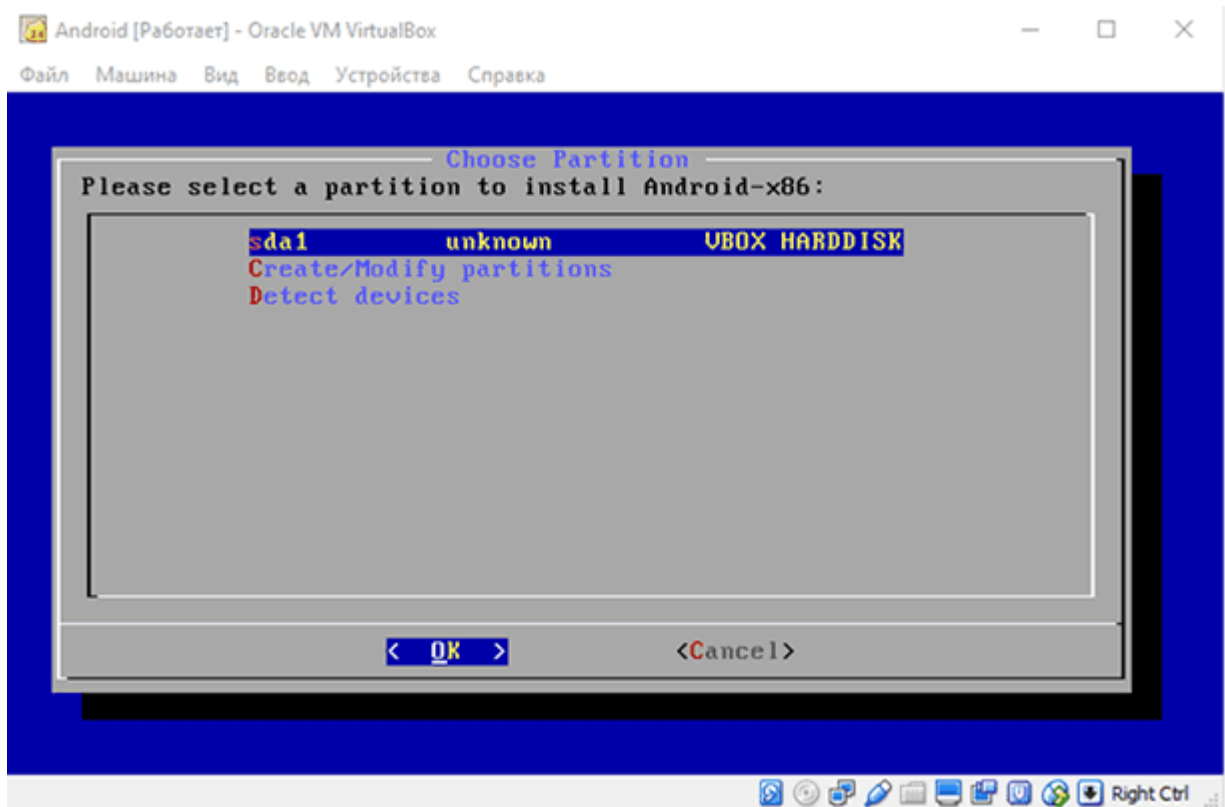
10. Затем запишите данные на диск используя опцию Write. Используйте стрелки на клавиатуре для выбора опции и Enter для применения. Затем подтвердите выбор напечатав Yes и нажав Enter.



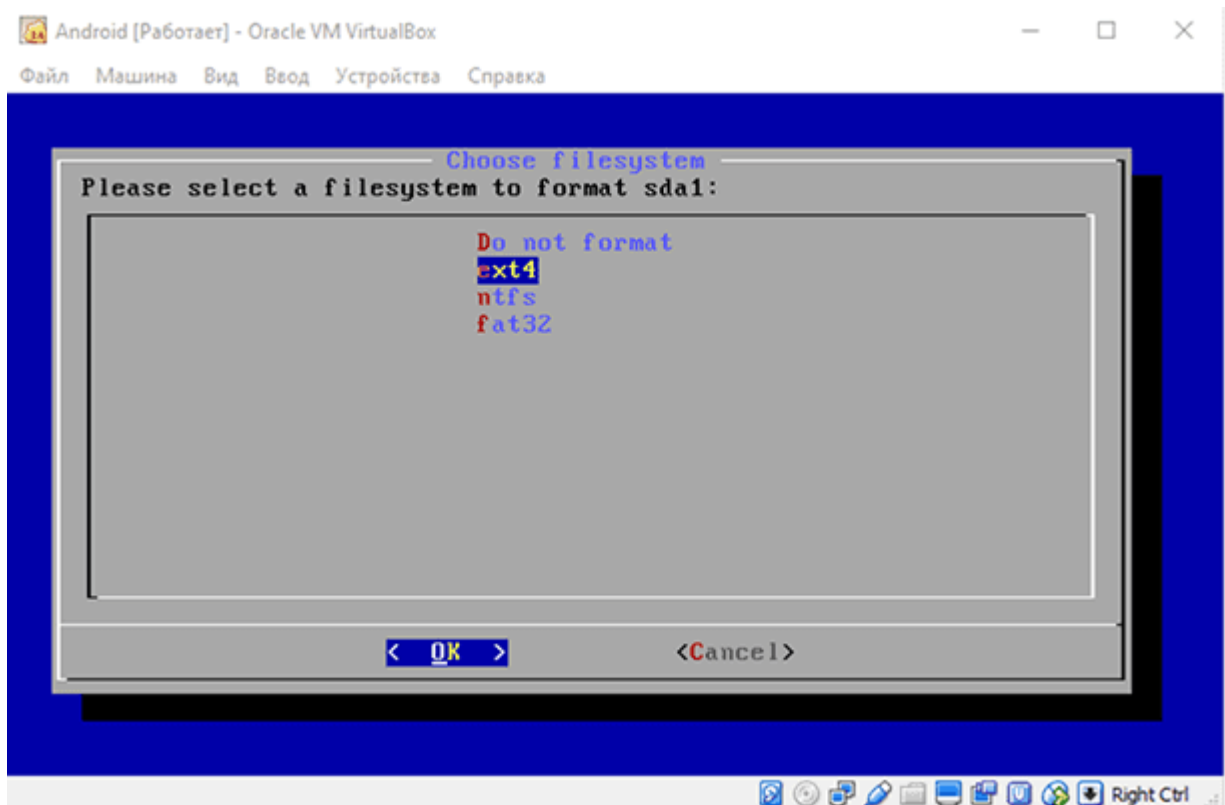
11. После завершения процедуры выйдите из утилиты выбрав Quit и нажав Enter.



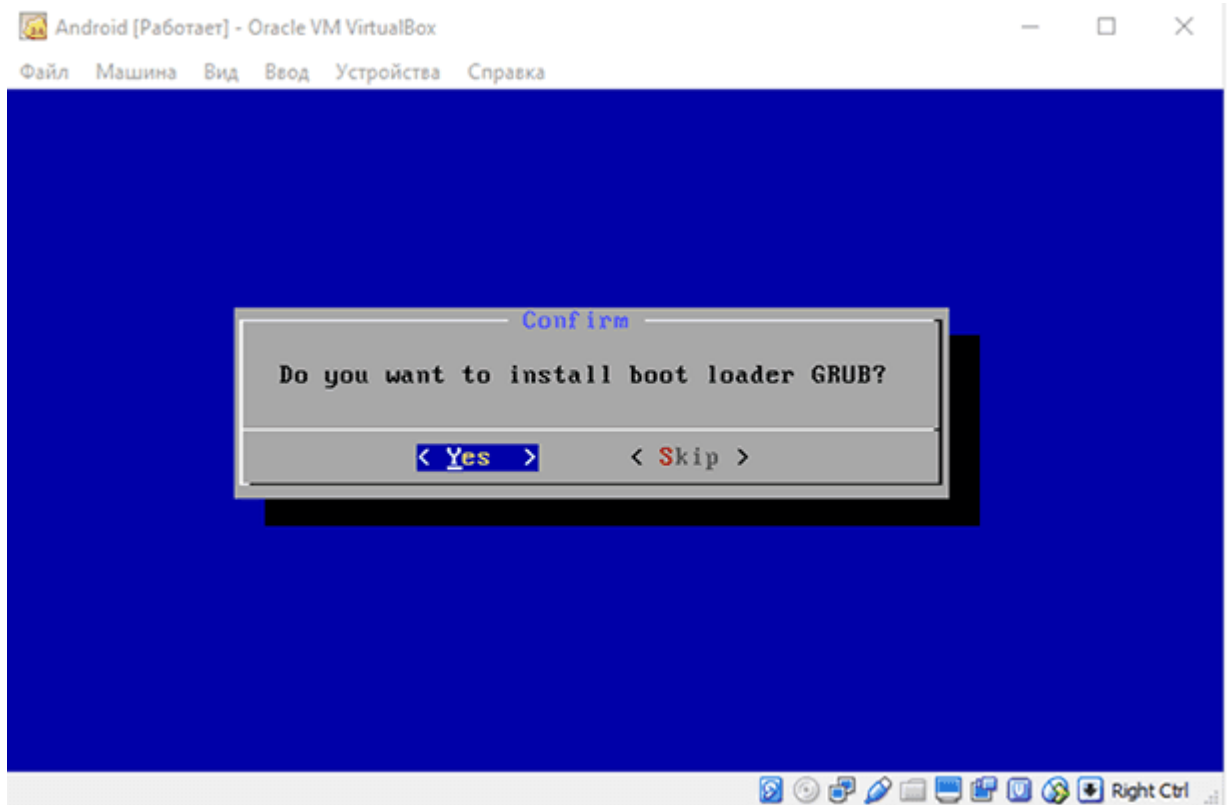
12. На следующем шаге выберите только что созданный раздел для установки Android и нажмите Enter.



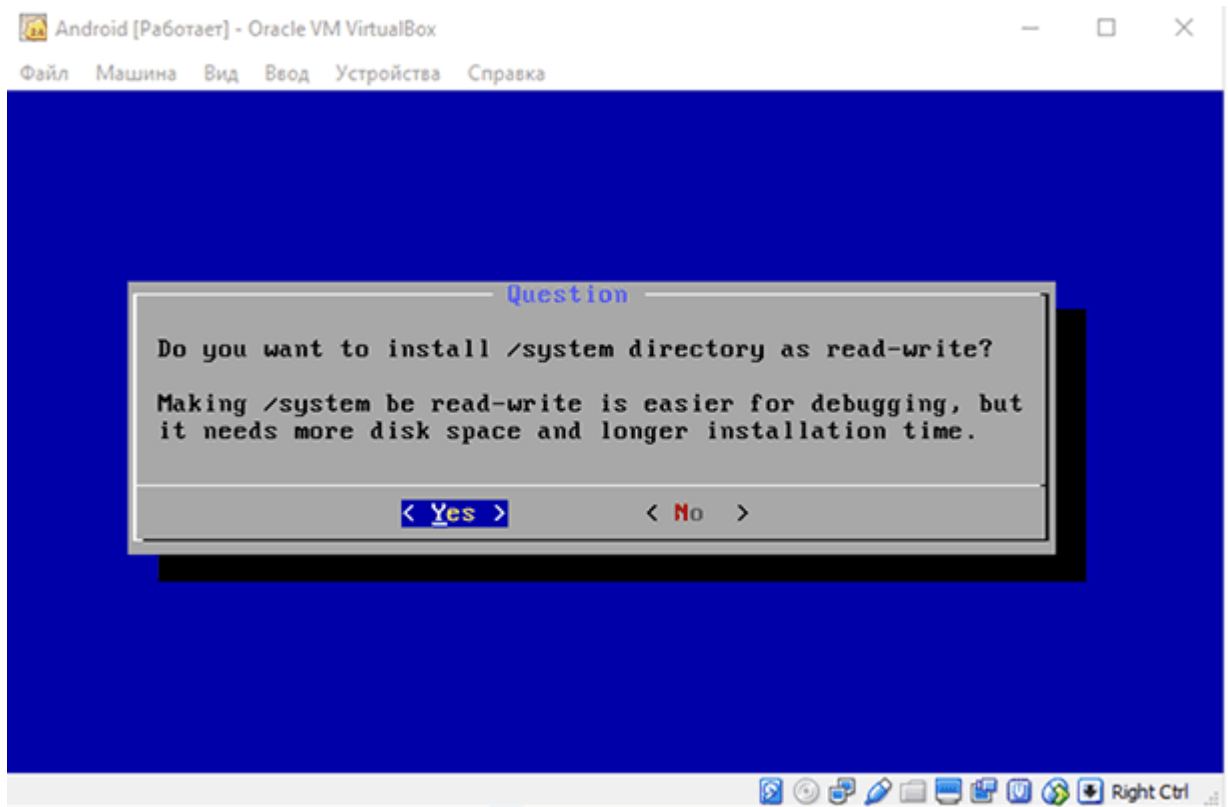
13. На следующем шаге укажите тип файловой системы Ext4 и нажмите Enter для подтверждения. Подтвердите ваш выбор и дождитесь окончания процесса форматирования.



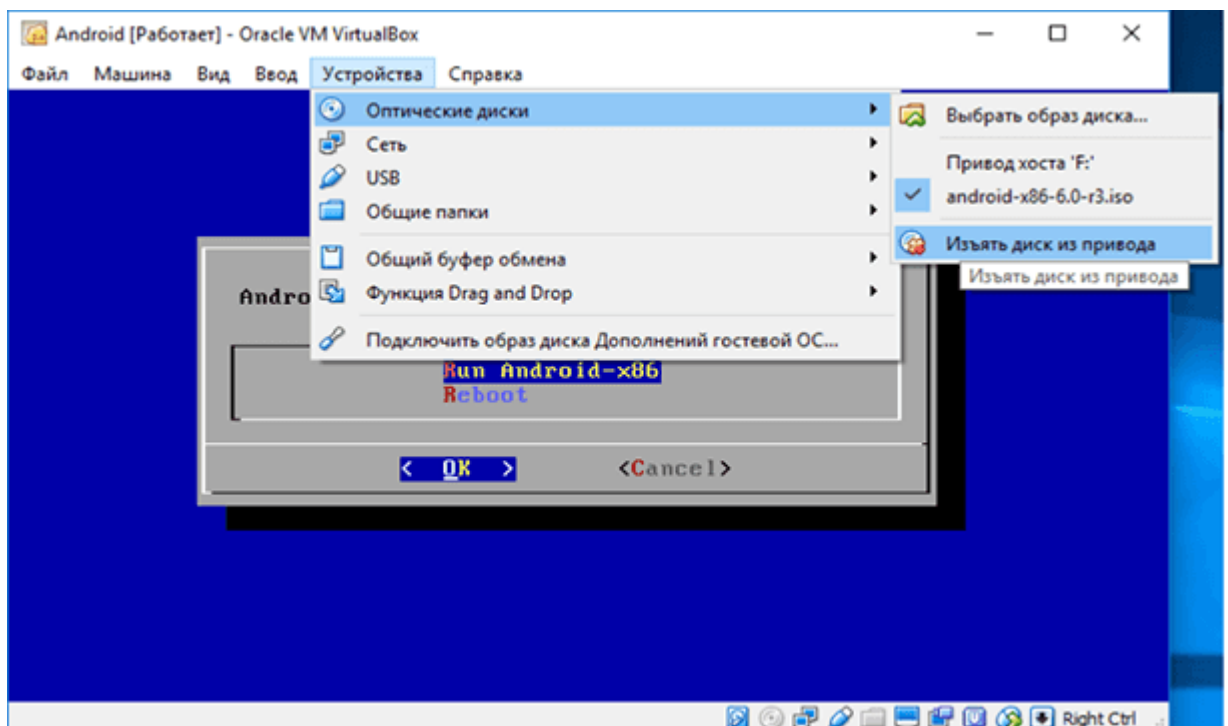
14. Подтвердите установку загрузчика GRUB выбрав Yes и нажав Enter.



15. Подтвердите желание сделать системную директорию доступную для чтения и записи. Выберите Yes и нажмите Enter, затем дождитесь установки Android.



16. На последнем шаге вы можете сразу загрузиться в Android или перезагрузить виртуальную машину. Отключите инсталляционный ISO образ с которого вы устанавливали виртуальную машину используя опцию главного меню Устройства – Оптические диски – Изъять диск из привода.

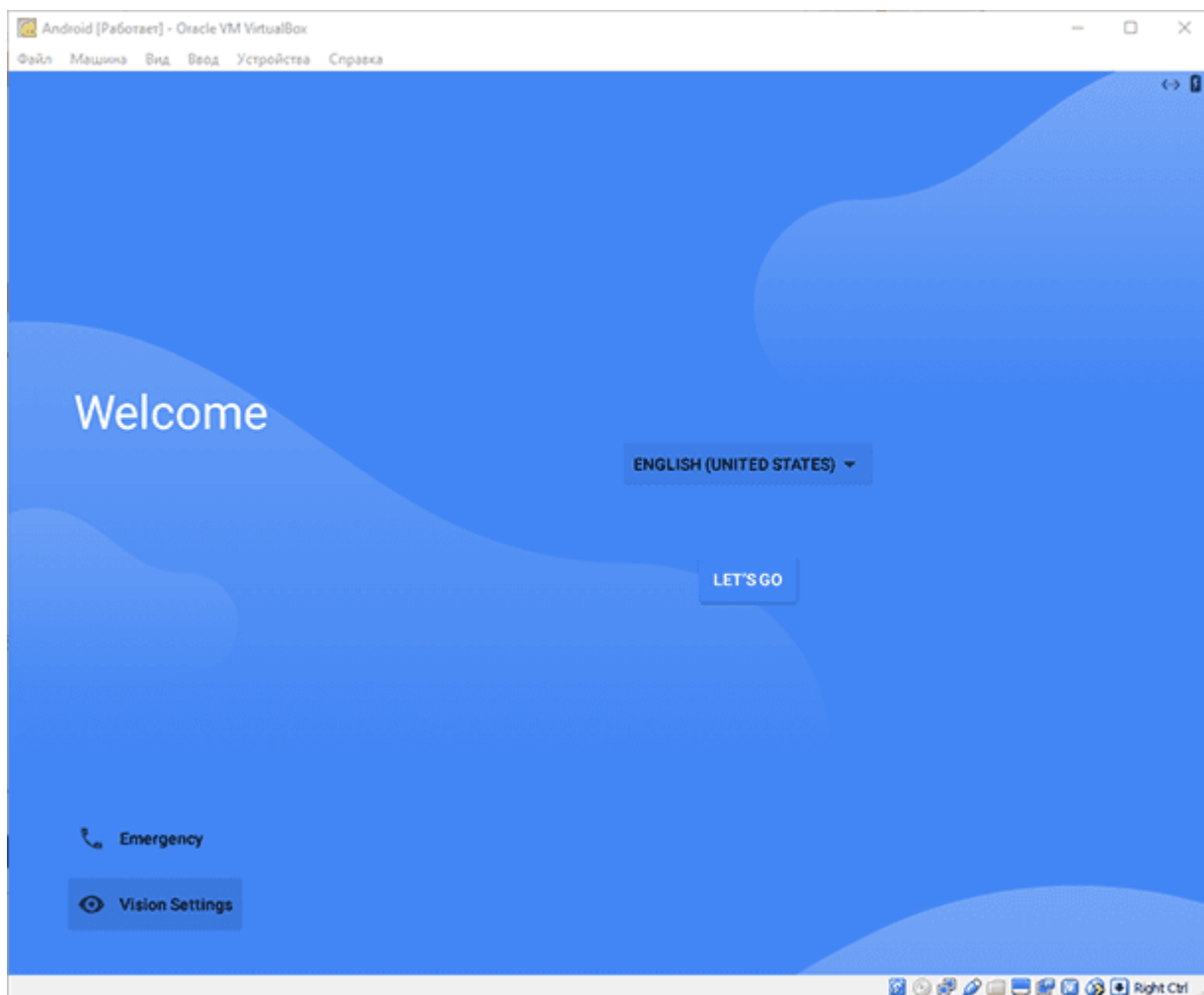


Если перезагрузить VirtualBox без отключения ISO образа, вы снова попадете в мастер установки.

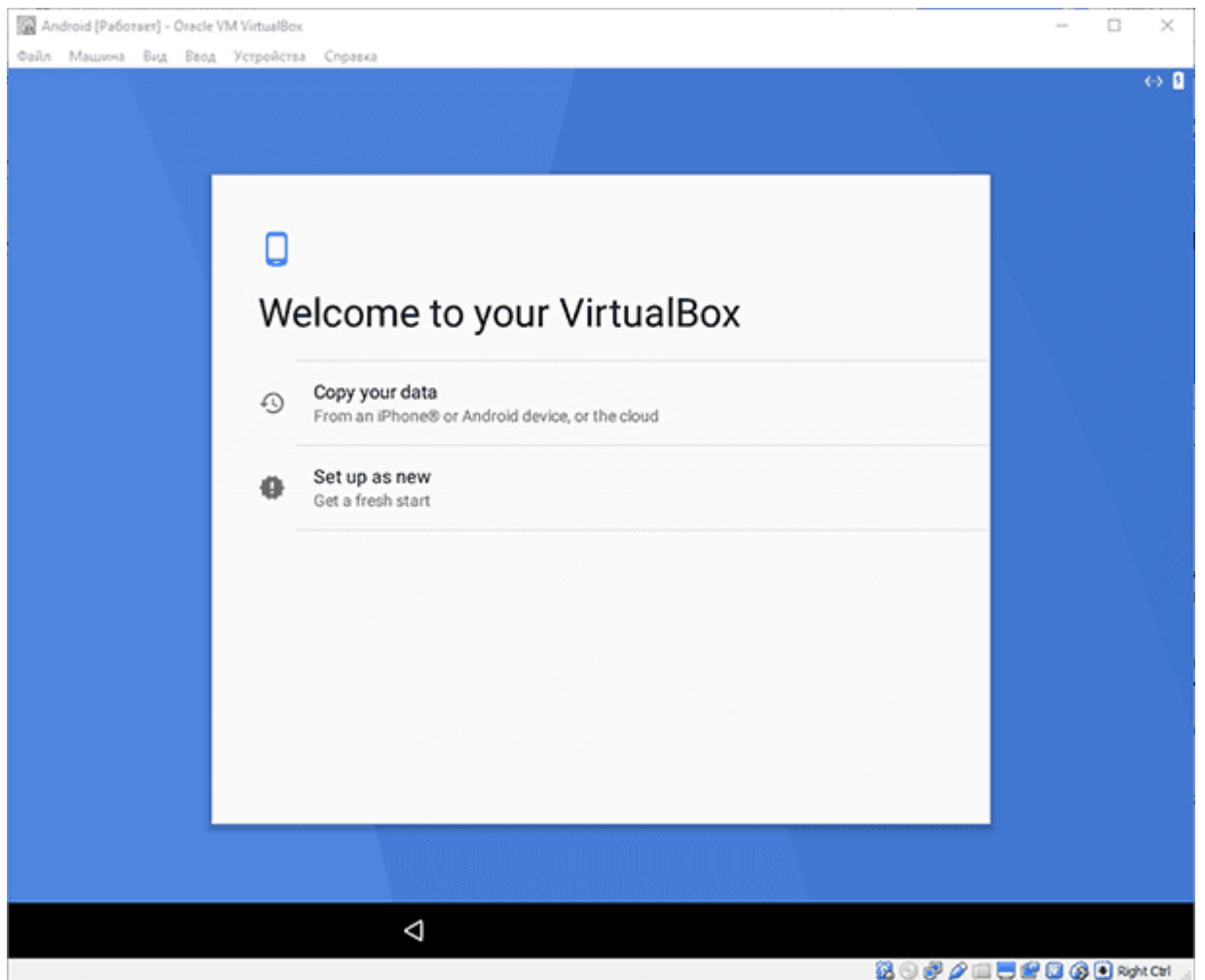
Настройка и работа с Android в VirtualBox

После перезагрузки VirtualBox выберите вариант загрузки по умолчанию и дождитесь полной загрузки системы. После первого старта системы вам необходимо будет настроить устройство:

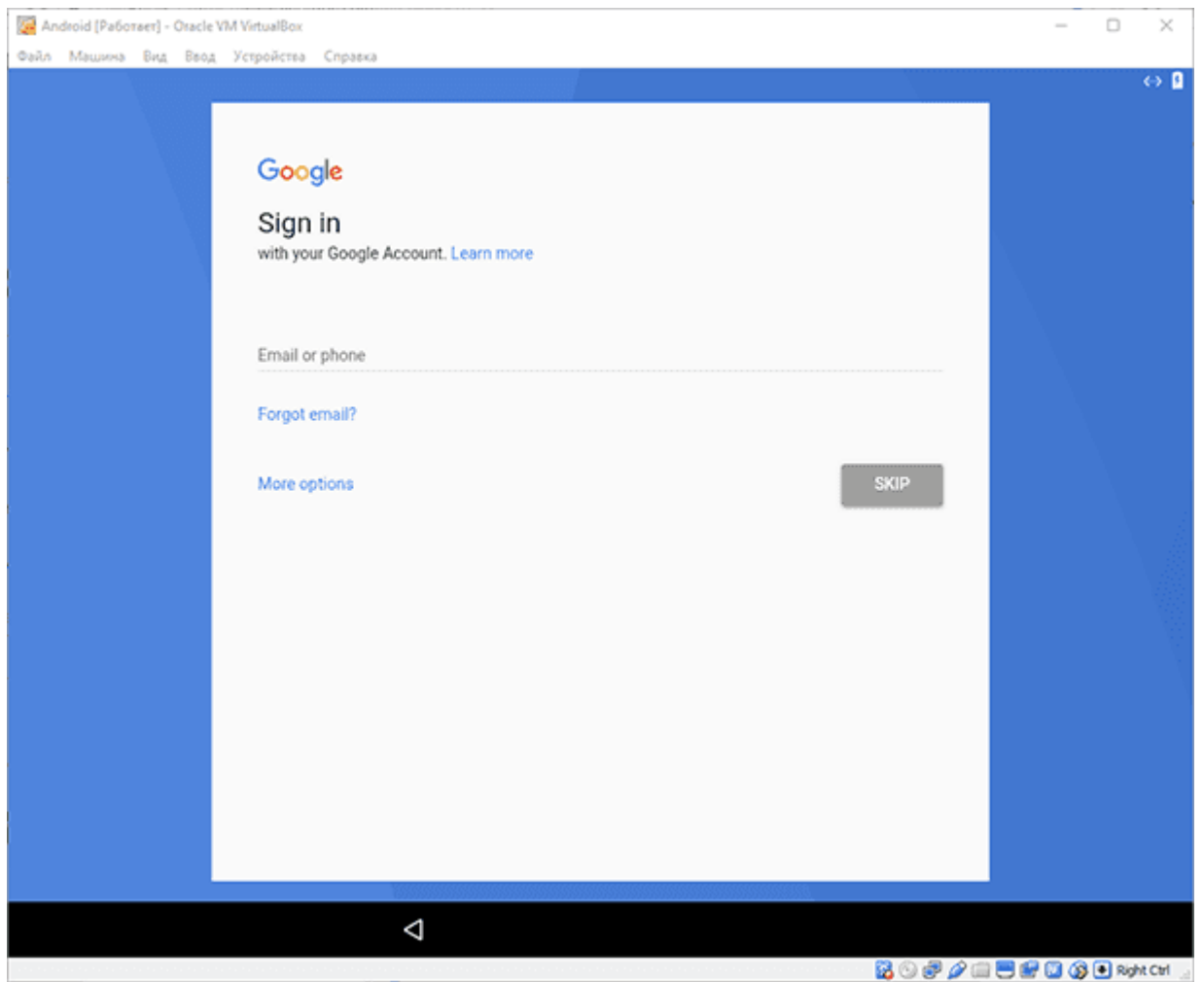
1. Укажите язык.



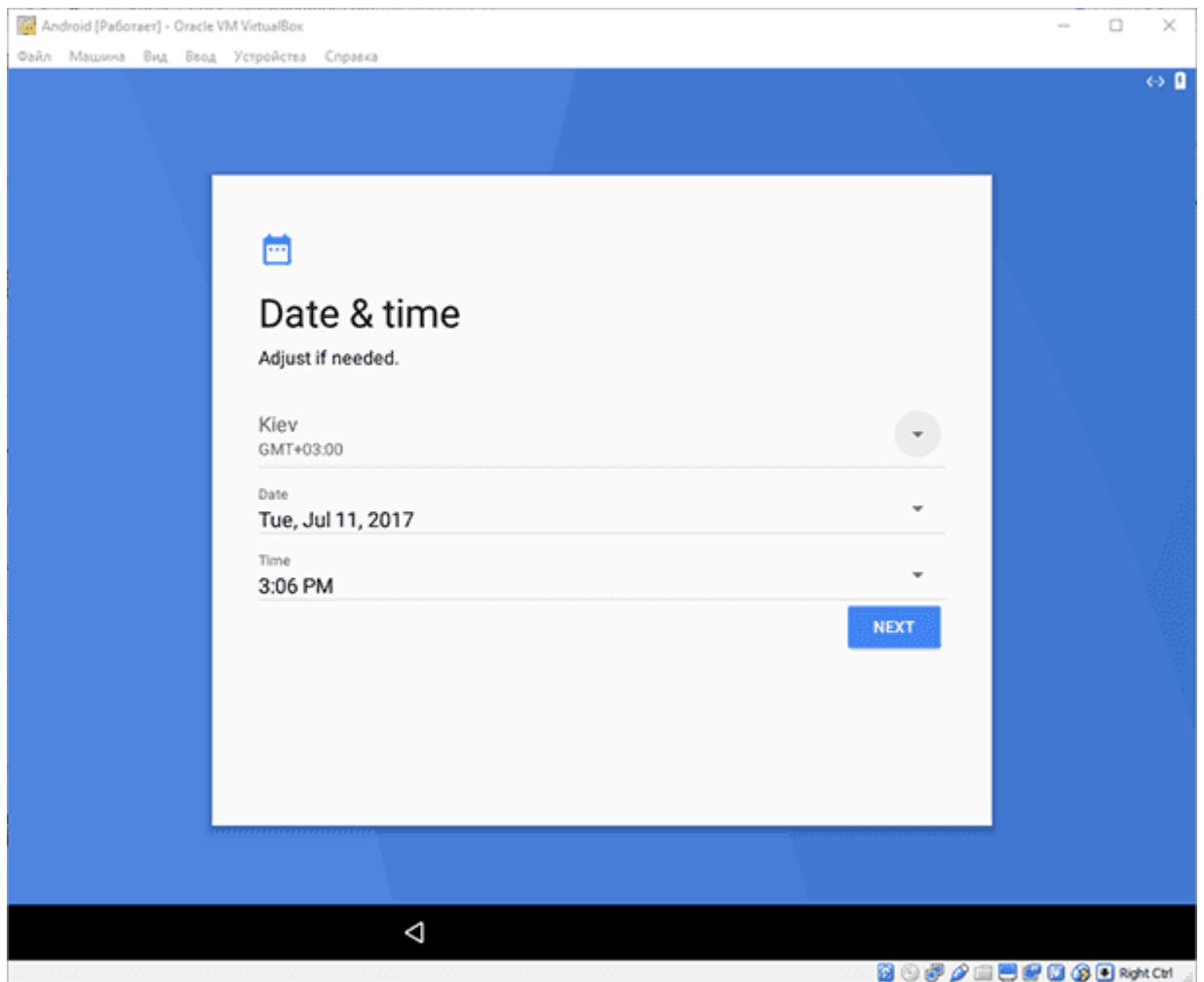
2. На шаге с предложением синхронизировать другое ваше Android устройство выберите – Set up as new.



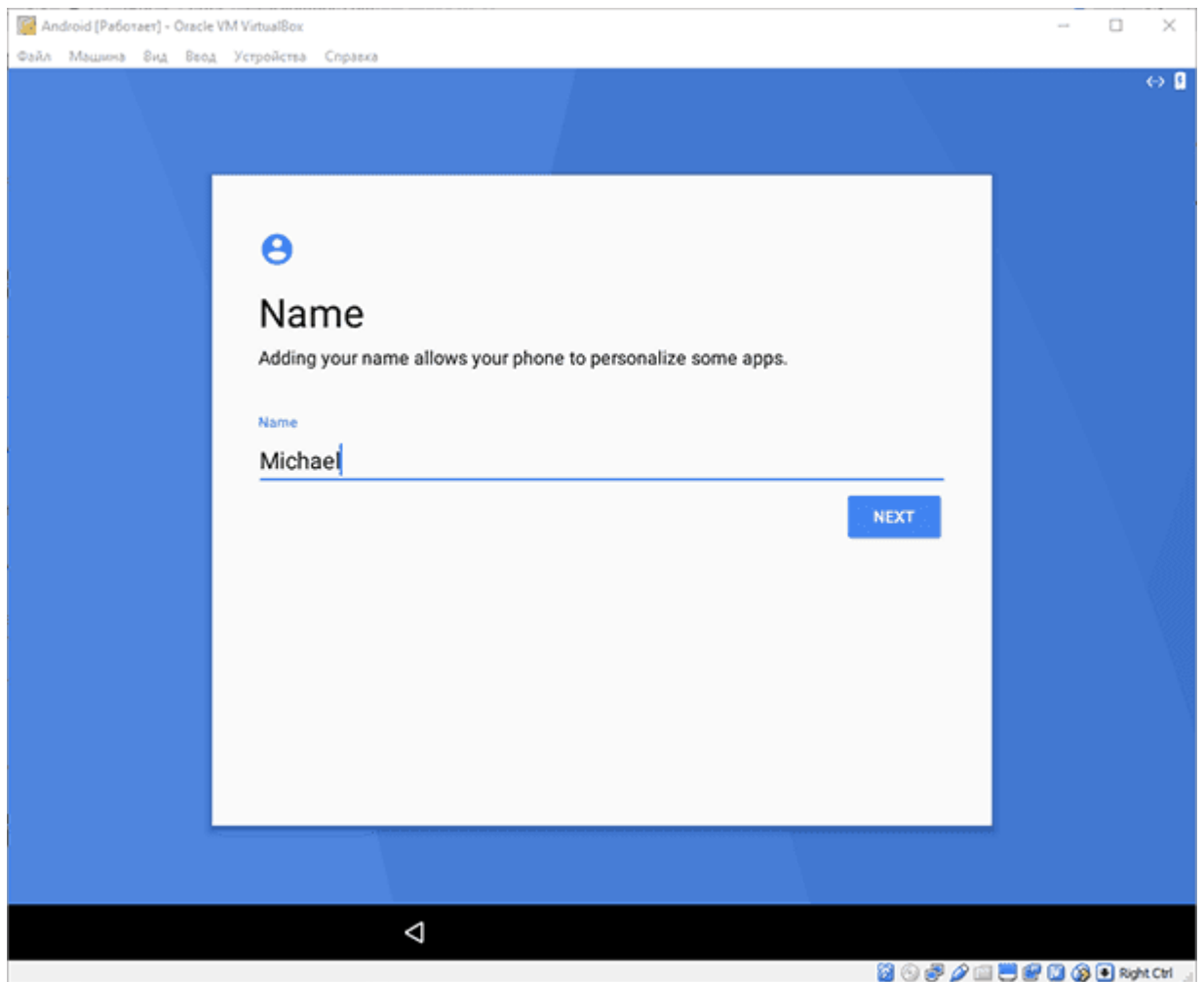
3. Пропустите шаг, на котором вам предлагают войти в Google account.



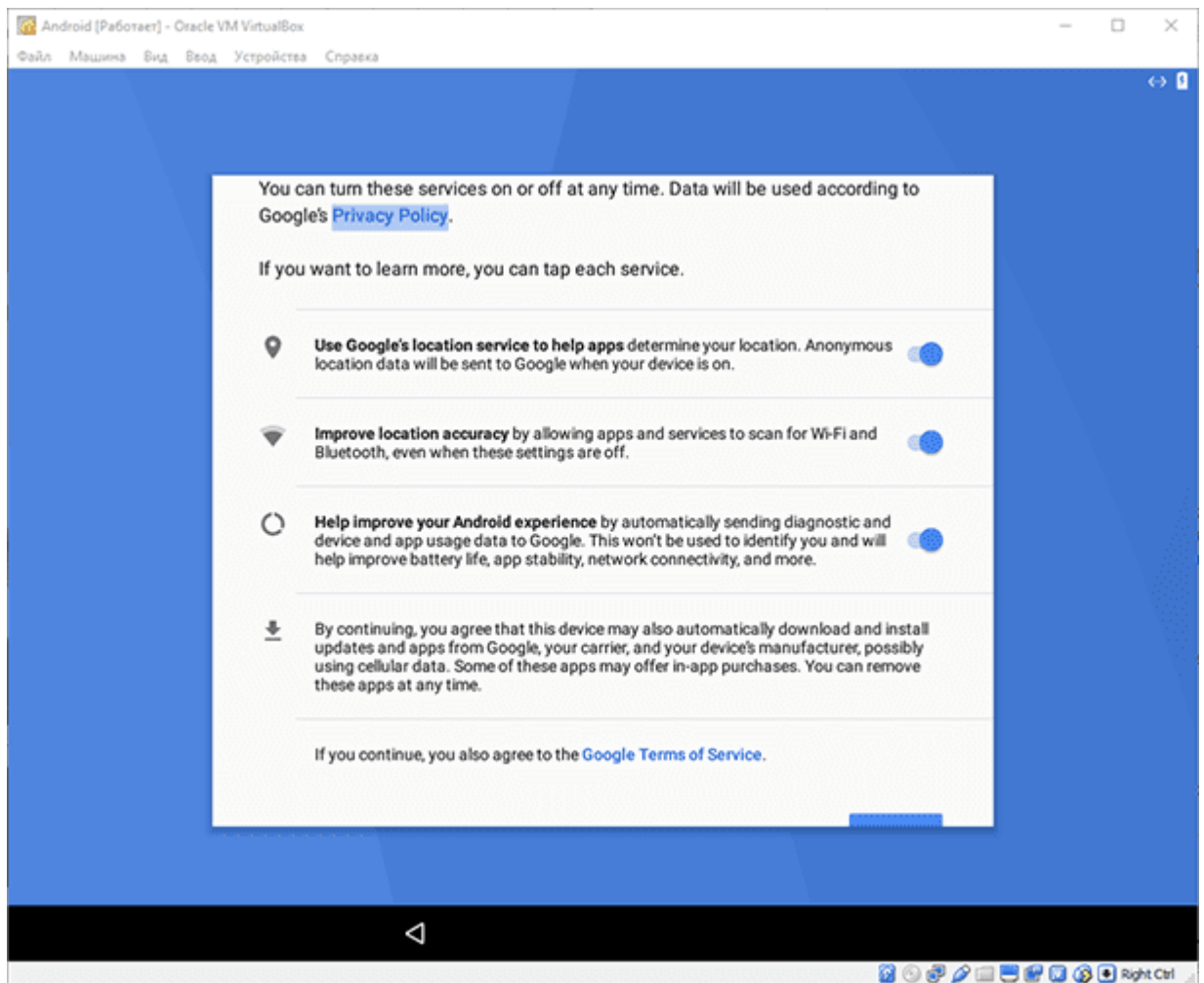
4. На следующем шаге установите дату и время:



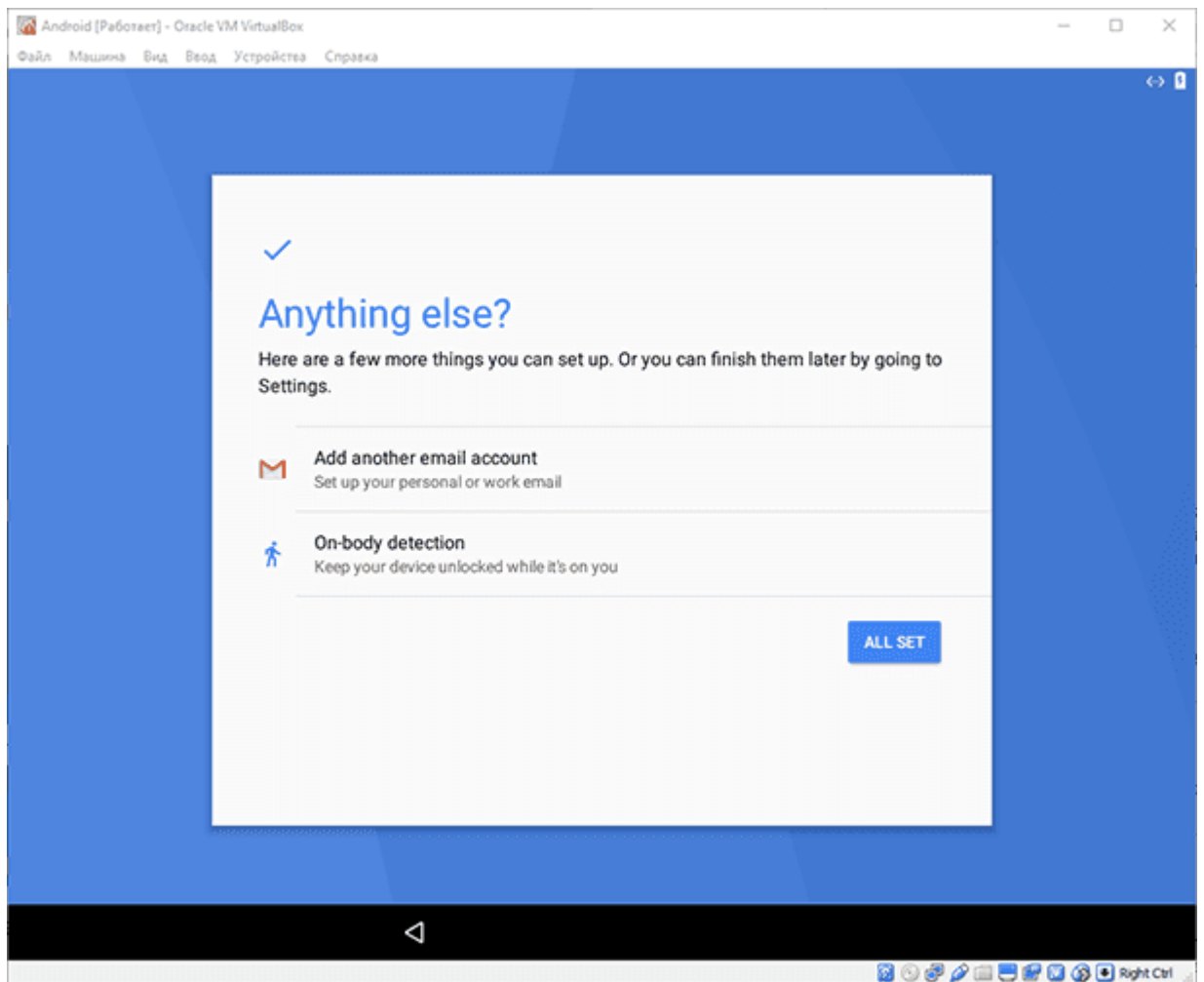
5. Введите ваше имя:



6. Прочитайте и примите пользовательское соглашение Google:



7. На последнем шаге нажмите All SET:



Использование виртуальной машины – не самый простой способ запуска Android приложения на Windows ПК. С помощью BlueStacks это можно сделать быстрее. Тем не менее с помощью VirtualBox вы получаете доступ к полноценной системе Android и можете экспериментировать с её настройками.

Библиографический список

1. Гостев И. М. Операционные системы: учебник и практикум для вузов / И. М. Гостев. – 2-е изд., испр. и доп. – М.: Изд-во Юрайт, 2021. – 164 с. Текст: электронный // ЭБС Юрайт [сайт]. – URL: <https://urait.ru/bcode/470010> (дата обращения: 16.06.2021).
2. Вицентий А. В., Рудина Е. С., Шишаев М. Г. Основы практической работы с UNIX-подобной операционной системой: Учебное пособие. Изд-во Мурманского арктического гос. ун-та. 2019. – 96 с.
3. Староверова Н. А. Операционные системы. Изд-во "Лань", 2021. – 412 с.

*Татьяна Алексеевна Галаган, доцент кафедры ИиУС АмГУ
(составитель)*

*Евгений Васильевич Дегтярев, преподаватель кафедры ИиУс АмГУ
(составитель)*

Методические указания к лабораторным работам по дисциплине «Операционные системы». Учебно-методическое пособие
