

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**



**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО  
ОБРАЗОВАНИЯ «ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ  
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Институт

компьютерных наук

Кафедра

автоматизированных систем управления

**ЛАБОРАТОРНАЯ РАБОТА №1**

По дисциплине "Операционные системы Linux"

На тему " Средства разработки программного обеспечения в ОС Linux "

Студент

ПИ-22-1

подпись, дата

Кистерёв В.А.

Руководитель

канд. техн. наук, доцент

ученая степень, ученое звание

подпись, дата

Кургасов В.В.

Липецк, 2024 г.

## **Оглавление**

Задание .....	3
Цель работы .....	3
Ход работы.....	4
Вывод.....	6
Контрольные вопросы .....	6

### **Задание**

- 1) Установить гипервизор, изучить понятия: Виртуализация, гипервизоры, изучить создание виртуальных машин.
- 2) Установить 64 серверную ОС на выбор (Ubuntu, Debian).
- 3) Написать программу. Дан одномерный массив из N целочисленных элементов (тип int). Упорядочить массив по возрастанию элементов.

### **Цель работы**

Получить представление о технологиях и средствах разработки программного обеспечения в ОС Linux.

## Ход работы

Для работы с гостевой операционной системой Debian будет использоваться гипервизор VirtualBox (рисунок 1).

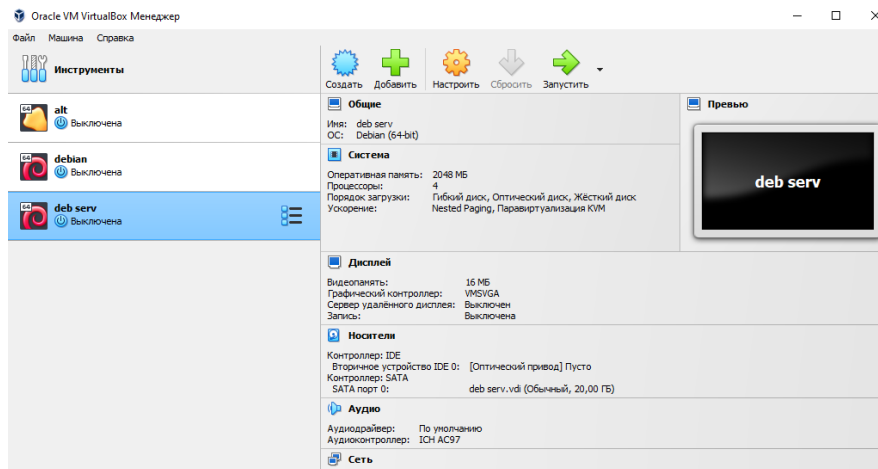


Рисунок 1 – Гипервизор VirtualBox

Выполним вход в учётную запись, созданную при установке ОС (рисунок 2).

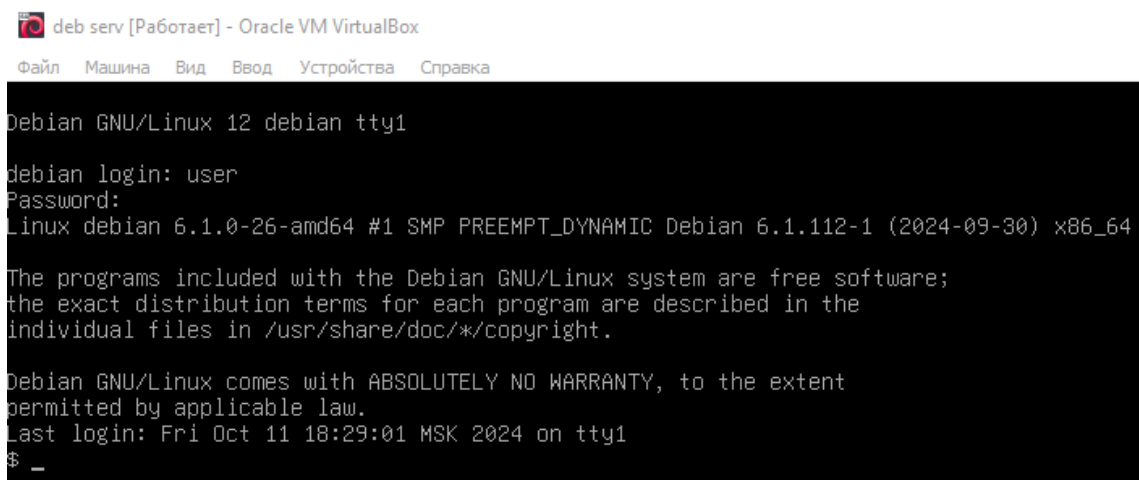


Рисунок 2 – Вход под пользователем user

Так как программа будет написана на языке Python, необходимо командой `apt install pip` установить необходимые пакеты (рисунок 3).

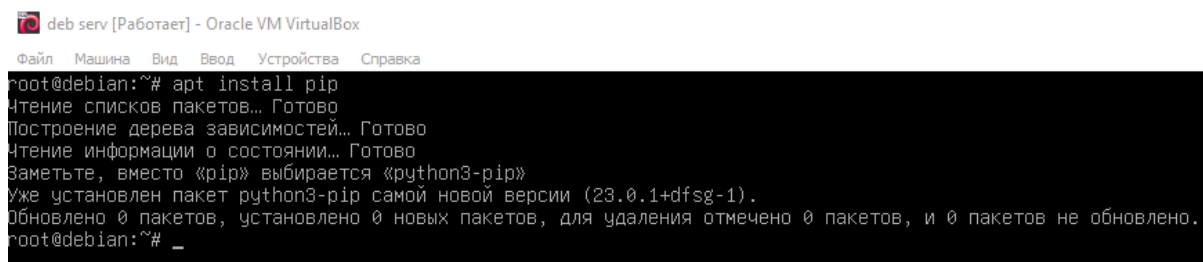
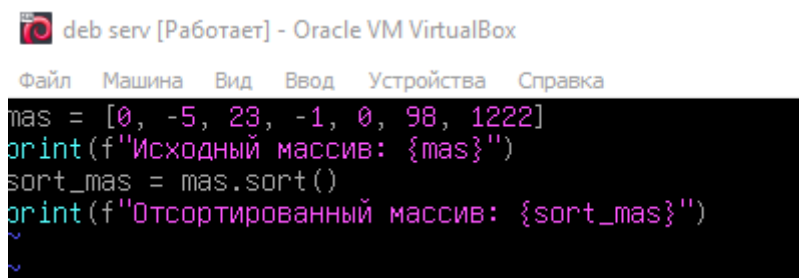


Рисунок 3 – Установка Python

С помощью редактора vim создадим новый файл main.py, который будет содержать программу для сортировки массива на языке Python. Программа в редакторе vim представлена на рисунке 4.

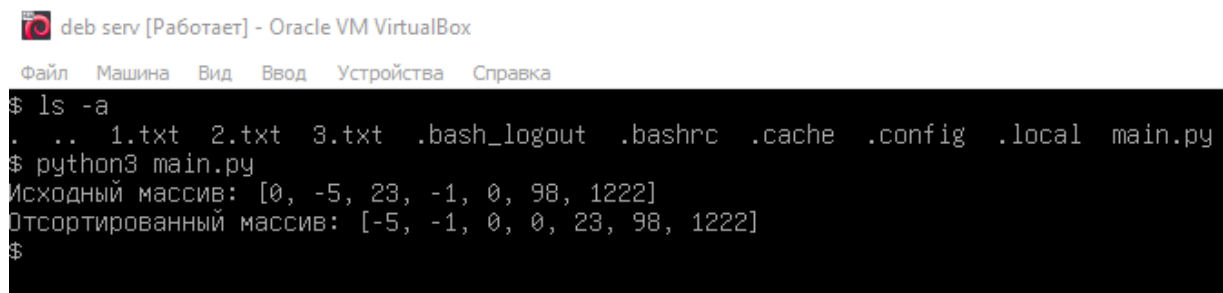


```
deb serv [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
mas = [0, -5, 23, -1, 0, 98, 1222]
print(f"Исходный массив: {mas}")
sort_mas = mas.sort()
print(f"Отсортированный массив: {sort_mas}")
```

Рисунок 4 – Программа в редакторе vim

Для сохранения файла и выхода из редактора переходим в режим команд и пишем «:wq».

Командой python3 выполним файл main.py, результат работы программы представлен на рисунке 5.



```
deb serv [Работает] - Oracle VM VirtualBox
Файл  Машина  Вид  Ввод  Устройства  Справка
$ ls -a
.  ..  1.txt  2.txt  3.txt  .bash_logout  .bashrc  .cache  .config  .local  main.py
$ python3 main.py
Исходный массив: [0, -5, 23, -1, 0, 98, 1222]
Отсортированный массив: [-5, -1, 0, 0, 23, 98, 1222]
$
```

Рисунок 5 – Результат работы программы

## **Вывод**

Таким образом, в ходе работы были получены представления о технологиях и средствах разработки программного обеспечения в ОС Linux.

## **Контрольные вопросы**

### **1) Что такое IDE?**

IDE (Integrated Development Environment) — это интегрированная среда разработки, которая включает в себя набор инструментов для программирования, таких как текстовый редактор, компилятор, отладчик и среда для сборки программы. Примеры IDE: Visual Studio, IntelliJ IDEA, PyCharm.

### **2) Что такое API?**

API (Application Programming Interface) — это интерфейс прикладного программирования, представляющий собой набор функций и протоколов, позволяющих одной программе взаимодействовать с другой. API предоставляет стандартизированный доступ к функциям системы или библиотеки, абстрагируя детали их реализации.

### **3) Что такое библиотека в программировании?**

Библиотека — это набор готовых функций, классов и процедур, которые могут быть повторно использованы в различных программах. Библиотеки содержат часто используемый код, который можно вызывать в других проектах для выполнения типичных задач (например, математические вычисления, работа с файлами).

### **4) Понятия статической и динамической библиотек.**

Статическая библиотека (Static Library, файл с расширением .lib или .a) компилируется и связывается с программой на этапе сборки. Код библиотеки встраивается в исполняемый файл программы, поэтому программа может быть запущена без дополнительных файлов.

Динамическая библиотека (Dynamic Library, расширение .dll или .so) подгружается в память во время выполнения программы. Программа использует динамическую библиотеку через ссылки на неё, что уменьшает размер

исполняемого файла и позволяет обновлять библиотеку без перекомпиляции программы.

5) Что такое плагин?

Плагин — это дополнительный программный модуль, который расширяет функциональность основной программы. Плагины обычно используются для добавления новых функций без изменения исходного кода программы. Примеры — плагины для текстовых редакторов, браузеров и медиаплееров.

6) Назовите несколько консольных текстовых редакторов для Linux.

Vim, Nano, Emacs, MCEdit, joe.

7) Что делает команда gcc?

Команда gcc (GNU Compiler Collection) используется для компиляции исходного кода на языках программирования, таких как C, C++, Objective-C и других, в исполняемые файлы или объектные файлы.

8) Что делает команда make?

Команда make запускает процесс сборки программы в соответствии с правилами, описанными в файле Makefile. Этот файл содержит инструкции по компиляции и сборке программы, что особенно полезно в проектах с большим количеством файлов исходного кода.

9) Что делает команда gdb?

gdb — это отладчик программ для GNU. С помощью gdb можно запускать программы в пошаговом режиме, устанавливать точки останова, исследовать состояние памяти и переменных, отслеживать выполнение кода для нахождения и исправления ошибок.

10) Дайте определение заголовочного файла и файла реализации.

Заголовочный файл (header file, расширение .h) содержит объявления функций, классов и переменных, но не их реализацию. Он служит интерфейсом между различными единицами трансляции (например, исходными файлами).

Файл реализации (implementation file, расширение .c или .cpp) содержит непосредственно реализацию функций и классов, объявленных в заголовочном файле.

11) Что означает единица трансляции? В чем особенность разработки программ из нескольких единиц трансляции?

Единица трансляции — это один файл исходного кода и все файлы, которые подключены к нему через директивы `#include`. Когда компилятор компилирует программу, каждая единица трансляции компилируется независимо от других.

Особенность разработки из нескольких единиц трансляции заключается в том, что это улучшает структуру кода, уменьшает время компиляции и упрощает поддержку программ, состоящих из множества модулей. Однако важно следить за тем, чтобы корректно организовать взаимодействие между единицами через заголовочные файлы и избегать проблем множественного включения.