



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ЛИПЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Институт
Кафедра

Компьютерных наук
Прикладной математики

ЛАБОРАТОРНАЯ РАБОТА

По дисциплине: «Операционные системы Linux».
На тему: «Средства разработки программного обеспечения в ОС Linux».

Студент

ПМ-22

группа

подпись, дата

Борисов А. В.

фамилия, инициалы

Руководитель

К. Т. Н.

ученая степень, ученое звание

подпись, дата

Кургасов В. В.

фамилия, инициалы

Липецк 2024

Цель

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

Ход работы

1. Установить редактор Vim, провести настройку и установить не менее 5 плагинов.
2. Написать программу с назначенным вариантом используя предложенный редактор.

1. Установка Vim:

```
exerted@exerte:~$ sudo apt install vim
```

Установка плагинов:

```
call plug#begin('~/.vim/plugged')
Plug 'neoclide/coc.nvim', {'branch': 'release'}
Plug 'preservim/nerdtree'
Plug 'shreerun/vim-polyglot'
plug 'tpope/vim-sensible'
Plug 'tpope/vim-commentary'
call plug#end()
```

2. Установка Cargo и метапакета build-essential который отвечает за компиляцию программного обеспечения из исходного кода.

```
exerted@exerte:~$ sudo apt install curl build-essential
```

Установщик Rust:

```
exerted@exerte:~$ sudo curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

Вариант 1:

Упорядочить массив по убыванию:

```
use std::io;

fn main() {
    println!("Введите массива:");
    let mut input: String = String::new();
    io::stdin().read_line(&mut input).unwrap();
    let mut vec: Vec<i32> = input.trim().split(' ').map(|str| str.parse::<i32>().unwrap()).collect::<Vec<i32>>();
    vec.sort_by(|a, b| b.cmp(a));
    println!("Отсортированный массив: {:?}", vec);
}
```

```
exerted@exerte:~/app$ cargo run
    Finished `dev` profile [unoptimized + debuginfo] target(s) in 0.31s
    Running `target/debug/app`
Введите массива:
1 2 3 4 5 6 32 4 5 2
Отсортированный массив: [32, 6, 5, 5, 4, 4, 3, 2, 2, 1]
```

Вывод:

В ходе лабораторной работы была разработана и протестирована программа сортировки элементов массива на языке Rust. Для написания программы был использован текстовый редактор vim, а компиляция выполнена с помощью утилиты rustc. Было получено представление о разработке на системе Linux.

Контрольные вопросы:

1. Что такое IDE?

IDE (Integrated Development Environment) — это интегрированная среда разработки, которая предоставляет разработчикам инструменты для написания, редактирования, компиляции и отладки кода.

2. Что такое API?

API (Application Programming Interface) — это набор методов, протоколов и инструментов, которые дают возможность одной программе взаимодействовать с другой.

3. Что такое библиотека в программировании?

Библиотека — это набор заранее написанных и скомпилированных функций или классов.

4. Понятия Статической и Динамической библиотеки.

Статическая библиотека — это библиотека, которая встраивается в исполняемый файл программы во время компиляции.

Динамическая библиотека — это библиотека, которая загружается во время выполнения программы.

5. Что такое плагин?

Плагин — это расширение или модуль, который добавляет новую функциональность в существующую программу.

6. Назовите несколько консольных текстовых редакторов для Linux.

Примеры консольных текстовых редакторов: vim, nano, emacs.

7. Что делает команда gcc?

gcc (GNU Compiler Collection) — это компилятор, который преобразует исходный код на C, C++ и других языках в исполняемые программы.

8. Что делает команда make?

make — это утилита для автоматизации сборки программ. Она читает файл Makefile, который описывает зависимости и правила сборки.

9. Что делает команда gdb?

gdb (GNU Debugger) — это отладчик, который позволяет разработчикам анализировать и исправлять ошибки в программах, отлаживать программы и следить за их выполнением.

10. Дайте определение заголовочного файла и файла реализации.

Приведите пример.

- Заголовочный файл (.h) содержит объявления функций, типов данных, макросов и переменных, доступные для использования в других частях программы.

- Файл реализации (.c или .cpp) содержит реализацию функций, которые были объявлены в заголовочном файле.

Пример: math.h (заголовочный файл) и math.c (файл реализации).

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

Единица трансляции — это отдельный файл исходного кода и его заголовочные файлы, которые компилируются в один объектный файл. Программы могут состоять из нескольких единиц трансляции, что упрощает разделение кода и повторное использование модулей.

12. Дайте краткую характеристику каждому этапу трансляции программ, написанных на Си.

Этапы трансляции программы на Си:

- Препроцессинг — обработка макросов и директив компилятора (#include, #define).

- Компиляция — преобразование исходного кода в ассемблерный код.
- Ассемблирование — преобразование ассемблерного кода в машинный код (объектные файлы).
- Линковка — связывание объектных файлов и библиотек для создания исполняемого файла.

