

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ**

**ОТЧЕТ
по лабораторной работе №1
по дисциплине «Программирование»
Тема: Регулярные выражения. Вариант 2**

Студент гр. 3343

Жучков О.Д.

Преподаватель

Государкин Я.С.

Санкт-Петербург

2024

Цель работы

Цель работы заключается в разработке программы на языке программирования C для поиска нужных строк, используя регулярные выражения.

Задание

На вход программе подается текст, представляющий собой набор предложений с новой строки. Текст заканчивается предложением "Fin." В тексте могут встречаться примеры запуска программ в командной строке Linux. Требуется, используя регулярные выражения, найти только примеры команд в оболочке суперпользователя и вывести на экран пары <имя пользователя> - <имя_команды>. Если предложение содержит какой-то пример команды, то гарантируется, что после нее будет символ переноса строки.

Примеры имеют следующий вид:

- Сначала идет имя пользователя, состоящее из букв, цифр и символа _
- Символ @
- Имя компьютера, состоящее из букв, цифр, символов _ и -
- Символ : и ~
- Символ \$, если команда запущена в оболочке пользователя и #, если в оболочке суперпользователя. При этом между двоеточием, тильдой и \$ или # могут быть пробелы.
- Пробел
- Сама команда и символ переноса строки.

Выполнение работы

Программа принимает набор строк и осуществляет их обработку. В соответствии с заданным регулярным выражением выполняется проверка введенных символов каждой строки. В процессе работы программы используются библиотеки для работы с регулярными выражениями, строками и стандартным вводом-выводом.

Вывод

Был написан код на языке C, выполняющий проверку строк соответственно регулярным выражениям. Для решения задачи была изучена и применена библиотека для работы с регулярными выражениями.

ПРИЛОЖЕНИЕ А ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: menu.c

```
#include <stdlib.h>
#include <stdio.h>
#include <string.h>
#include <regex.h>

int main() {
    char * ex = "([a-zA-Z0-9_]+)@[a-zA-Z0-9_-]+: *~ *# (.+)";
    regex_t regex;
    regmatch_t match[3];
    regcomp(&regex, ex, REG_EXTENDED);
    char line[201];
    do {
        fgets(line, 200, stdin);
        if (regexec(&regex, line, 3, match, 0) == 0) {
            printf("%.s - %.s", match[1].rm_so-match[1].rm_so,
                line+match[1].rm_so, match[2].rm_so-
match[2].rm_so,
                line+match[2].rm_so);
        }
    } while (strcmp(line, "Fin.") != 0);
    return 0;
}
```