

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

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

Студент гр. 3343

Никишин С.А.

Преподаватель

Государкин Я. С.

Санкт-Петербург

2024

Цель работы

Изучить функции библиотеки `regex.h` языка Си, освоить 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

Описание функций:

- `int main()`: главная функция программы, выводит все совпадения по заданному шаблону

Разработанный программный код см. в приложении А.

Тестирование

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	<p>This is simple url: http://www.google.com/track.mp3 May be more than one upper level domain http://www.google.com.edu/hello.avi Many of them. Rly. Look at this! http://www.qwe.edu.etu.yahooo.org.net.ru/qwe.q Some other protocols ftp://skype.com/qqwe/qweqw/qwe.avi Fin.</p>	<p>google.com - track.mp3 google.com.edu — hello.avi qwe.edu.etu.yahooo.org. net.ru - qwe.q skype.com - qwe.avi</p>	<p>Выходные данные соответствуют ожиданиям.</p>
2.	<p>This is simple url: http://www.google.-aaaaaa.com/track.mp3 May be more than one upper level domain http://www.google.com.edu/hello.avi Many of them. youtube.en/file.f Rly. Look at this! Fin.</p>	<p>google.-aaaaaa.com - track.mp3 google.com.edu - hello.avi youtube.en - file.f</p>	<p>Выходные данные соответствуют ожиданиям.</p>
3.	<p>This is simple url: http://www.google.aaaaaa.com//track.mp3 May be more than one upper level</p>	<p>google.com.edu - hello.avi youtube.en - file.f google_google.com.edu</p>	<p>Выходные данные соответствуют</p>

<p>domain</p> <p>http://www.google.com.edu/hello.avi</p> <p>Many of them. youtube.en/file.f</p> <p>Rly. Look at this! This is simple url:</p> <p>aaa://googleaaaaacom/a.a</p> <p>May be more than one upper level</p> <p>domain</p> <p>http://www.google_google.com.edu/hello.avi</p> <p>Fin.</p>	- hello.avi	ожиданиям.
---	-------------	------------

Выводы

В ходе выполнения лабораторной работы были освоены необходимые навыки для использования регулярных выражений на языке Си с помощью библиотеки `regex.h`.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

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

int main() {
    regex_t r;
    regcomp(&r, "(\\w+)@[-w]+: *~ *# (.+)", 1);
    char t[77];
    regmatch_t m[3];
    while (fgets(t, 77, stdin))
        if (!regexec(&r, t, 3, m, 0))
            printf("%.s - %.s", (int)(m[1].rm_eo - m[1].rm_so),
&t[m[1].rm_so], (int)(m[2].rm_eo - m[2].rm_so), &t[m[2].rm_so]);
    return 0;
}
```