

МИНОБРНАУКИ РОССИИ
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

Студент гр. 3344

Сербиновский Ю.М.

Преподаватель

Глазунов С.А.

Санкт-Петербург

2024

Цель работы

Научиться использовать регулярные выражения в языке программирования С.

Задание.

Вариант 2

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

Регулярное выражение записано в `reg_str`, данная строка компилируется, и результат записывается в `reg_compiled`. Далее в цикле из `stdin` считывается по одному предложению (каждое предложение — с новой строки). При помощи `regexec` каждое предложение проверяется на соответствие регулярному выражению. Если проверка пройдена, то при помощи двойного цикла `for` и массива структур `regmatch_t match_groups`, хранящего информацию о группах регулярного выражения относительно текущего предложения, в `stdout` выводится комбинация <имя пользователя> - <имя_команды>. Считывание завершается, когда пользователь вводит «Fin.».

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные
1.	<pre>Run docker container: kot@kot-ThinkPad:~\$ docker run -d -- name stepik stepik/challenge-avr:latest You can get into running /bin/bash command in interactive mode: kot@kot-ThinkPad:~\$ docker exec -it stepik "/bin/bash" Switch user: su : root@84628200cd19: ~ # su box box@84628200cd19: ~ \$ ^C Exit from box: box@5718c87efaa7: ~ \$ exit exit from container: root@5718c87efaa7: ~ # exit kot@kot-ThinkPad:~\$ ^C Fin.</pre>	<pre>root - su box root - exit</pre>

Выводы

Был получен опыт работы с регулярными выражениями на языке C, была написана программа, определяющая по строке в консоли Linux, каким пользователем и какая была выполнена команда.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.c

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

int main() {
    char* reg_str = "([a-zA-Z0-9_]+)@[a-zA-Z0-9_-]+: ?~ ?# (.\n)";
    regex_t reg_compiled;
    regcomp(&reg_compiled, reg_str, REG_EXTENDED);
    size_t max_groups = 3;
    regmatch_t match_groups[max_groups];

    char cur_str[1000];
    fgets(cur_str, 1000, stdin);
    while(strstr(cur_str, "Fin.") == 0) {
        if (regexexec(&reg_compiled, cur_str, max_groups, match_groups, 0)
== 0) {
            for (int i = 1; i < max_groups; i++) {
                for (int j = match_groups[i].rm_so; j <
match_groups[i].rm_eo; j++) {
                    printf("%c", cur_str[j]);
                }
                if (i == 1)
                    printf(" - ");
            }
            fgets(cur_str, 1000, stdin);
        }
        return 0;
    }
}
```