

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. 9382

Дерюгин Д.А.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2020

Цель работы.

В данной работе мы учимся пользоваться регулярными выражениями в языке Си.

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.

Подключаем библиотеки для работы с программой. Пишется регулярное выражение, которое ищет сроки, подходящие по условию задания. Если есть ошибки в написании регулярного выражения, то функция `regcomp` завершит работы программы. В цикле проверяем строчку. Если строка имеет вид “Fin”, цикл прекращается. В самом же цикле при помощи функции `regexec()` проверяется, подходит ли данная строка под описания регулярного выражения. Если предложение подошло под регулярное выражения, выводим имя пользователя и имя команды через “ - ”. В конце освобождаем память .

Выводы.

В ходе лабораторной работы были получены навыки использования регулярного выражения.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

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

int main(){
    char* sentence = malloc(100*sizeof(char));
    char* regex = "([A-Za-z0-9_]+)@[A-Za-z0-9_-]+: ?~ ?# (.+)";
    size_t max_group = 3;
    regex_t compRegex;
    regmatch_t groupArray[max_group];
    if (regcomp(&compRegex, regex, REG_EXTENDED)){
        return 0;
    }
    while (strcmp(sentence, "Fin.")!=0){
        fgets(sentence, 100, stdin);
        if (regexexec(&compRegex, sentence, max_group, groupArray, 0)
==0){
            for(int i = 1; i<max_group; i++) {
                for (int j = groupArray[i].rm_so; j< groupArray[i].rm_eo;j++)
printf("%c", sentence[j]);
                if (i == 1) printf(" - "); }
            }
        free(sentence);
        regfree(&compRegex);
        return 0; }
```