

Требования к программам

1. Программа должна получать все параметры в качестве аргументов командной строки.

2. Аргументы командной строки для задач 1–5:

- 1) f_{in} – имя входного файла,
- 2) f_{out} – имя выходного файла,
- 3) s – строка s .
- 4) t – строка t .

Например, запуск

```
./a01.out a.txt b.txt "abcd" "(){ }[] ;+-*/*="
```

означает, что требуется читать строки из файла `a.txt`, выводить результат в файл `b.txt`, параметр-строка s равен `"abcd"`, параметр-строка t равен `"(){ }[] ;+-*/*="`.

3. Результатом работы каждой функции является измененный файл (не выводится в `main`) и возвращаемое значение (выводится в `main`).

4. Вывод результата работы функции в функции `main` должен производиться по формату:

```
printf ("%s : Task = %d Result = %d Elapsed = %.2f\n",  
        argv[0], task, res, t);
```

где

- `argv[0]` – первый аргумент командной строки (имя образа программы),
- `task` – номер задачи (1–8),
- `res` – результат работы функции, реализующей решение этой задачи,
- `t` – время работы функции, реализующей решение этой задачи.

Вывод должен производиться в точности в таком формате, чтобы можно было автоматизировать обработку запуска многих тестов.

Задачи

1. Написать функцию, получающую в качестве аргументов имя a текстового файла неизвестной длины, имя файла b для вывода информации и символьную строку s , и выводящую в файл b те строки файла a , которые совпадают со строкой s ; при этом символ '.' в строке s соответствует любому символу строки из файла a , символам '.' и '\' соответствуют последовательности символов '\.' и '\\\' в строке s . Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
2. Написать функцию, получающую в качестве аргументов имя a текстового файла неизвестной длины, имя файла b для вывода информации и символьную строку s , и выводящую в файл b те строки файла a , которые совпадают со строкой s ; при этом символ '?' в строке s означает, что предыдущий символ строки s может учитываться 0 или 1 раз, символам '?' и '\' соответствуют последовательности символов '\?' и '\\\' в строке s . Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
3. Написать функцию, получающую в качестве аргументов имя a текстового файла неизвестной длины, имя файла b для вывода информации и символьную строку s , и выводящую в файл b те строки файла a , которые совпадают со строкой s ; при этом символ '+' в строке s означает, что предыдущий символ строки s может учитываться 1 или более раз, символам '+' и '\' соответствуют последовательности символов '\+' и '\\\' в строке s . Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
4. Написать функцию, получающую в качестве аргументов имя a текстового файла неизвестной длины, имя файла b для вывода информации и символьную строку s , и выводящую в файл b те строки файла a , которые совпадают со строкой s ; при этом символ '*' в строке s означает, что предыдущий символ строки s может учитываться 0 или более раз, символам '*' и '\' соответствуют последовательности символов '*' и '\\\' в строке s . Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
5. Написать функцию, получающую в качестве аргументов имя a текстового файла неизвестной длины, имя файла b для вывода информации и символьную строку s , и выводящую в файл b те строки файла a , которые совпадают со строкой s ; при этом последовательность '[$n-m$]' (n , m – символы) в строке s соответствует любому символу строки из файла a , имеющему код в диапазоне $n \dots m$, символам '[' и ']' и '\' соответствуют последовательности символов '\[', '\]' и '\\\' в строке s . Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..