

Требования к программам

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

2. Аргументы командной строки для задач 1–2:

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

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"`.

3. Аргументы командной строки для задач 3–7:

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

Например, запуск

```
./a03.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 `"(){}[] ;+-*/*="`.

4. Аргументы командной строки для задачи 8:

- 1) f_{in} – имя входного файла,
- 2) f_{out} – имя выходного файла,
- 3) s – строка s ,
- 4) l – строка l ,
- 5) r – строка r ,

Например, запуск

```
./a08.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 l равен `"<"`, параметр-строка r равен `">"`.

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

6. 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*, в которые входит строка *s*, причем прописные и соответствующие строчные латинские буквы считаются совпадающими. 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*, у которых конец строки совпадает со строкой *s* (например, при *s*="ab" должны быть выведены строки, завершающиеся символами "ab"). 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 символьные строки *s* и *t*, и выводящую в файл *b* строки файла *a*, заменяя каждое вхождение строки *s* на строку *t*. Функция возвращает количество измененных 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 символьные строки *s* и *t*, и выводящую в файл *b* те строки файла *a*, которые не совпадают со строкой *s*; при этом пробельные символы не учитываются (пробельным называется символ, содержащийся в строке *t*). 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 символьные строки *s* и *t*, и выводящую в файл *b* те строки файла *a*, которые состоят только из символов строки *s*; при этом пробельные символы не учитываются (пробельным называется символ, содержащийся в строке *t*). Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
6. Написать функцию, получающую в качестве аргументов имя *a* текстового файла неизвестной длины, имя файла *b* для вывода информации и символьные строки *s* и *t*, и выводящую в файл *b* те строки файла *a*, в которые входит строка *s*; при этом пробельные символы не учитываются (пробельным называется символ, содержащийся в строке *t*). Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
7. Написать функцию, получающую в качестве аргументов имя *a* текстового файла неизвестной длины, имя файла *b* для вывода информации и символьные строки *s* и *t*, и выводящую в файл *b* те строки файла *a*, в которые входит строка *s*, начиная с первого не пробельного символа (пробельным называется символ, содержащийся в строке *t*). Функция возвращает количество таких строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
8. Написать функцию, получающую в качестве аргументов имя *a* текстового файла неизвестной длины, имя файла *b* для вывода информации и символьные строки *s*, *l* и *r*, и выводящую в файл *b* строки файла *a*, и заменяя каждое вхождение строки *s* на строку, образованную строками *l*, *s*, *r* (например, при *s*="a", *l*("(" и *r*=")"), строка "abc" заменяется на "(a)bc"). Функция возвращает количество измененных строк или -1 , -2 и т.д., если она не смогла открыть файл, прочитать элемент и т.д..