

## Простые чтение и запись

Пример использования файлов для решения задачи обращения массива чисел.

В файле input.txt хранятся числа в следующем формате: сначала идёт число  $n$  – количество чисел, а затем идут  $n$  чисел. В файл output.txt нужно записать эти же числа в обратном порядке. Программа, которая решает эту задачу выглядит так:

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

int main()
{
    // Открываем 2 файла - один на чтение(read - r), другой на запись (write - w)
    FILE* in  = fopen("input.txt", "r");
    FILE* out = fopen("output.txt", "w");
    // Считываем число элементов
    int n;
    fscanf(in, "%d", &n);

    // Выделяем память
    int* arr = (int*)malloc(n*sizeof(int));
    // Считываем числа в массив arr
    for (int i = 0; i < n; ++i)
        fscanf(in, "%d", &arr[i]);
    // Записываем числа в файл output.txt
    for (int i = 0; i < n; ++i)
        fprintf(out, "%d ", arr[n-i-1]);

    free(arr);
    fclose(in);
    fclose(out);
}
```

- **Reverse** Протестируйте программу выше. Не забудьте создать файл input.txt и заполнить его числами. (Эту задачу присылать не нужно)
- **Сумма чисел** В файле input.txt лежат числа (сначала идёт число  $n$  – количество чисел, а затем идут  $n$  чисел). Вам нужно записать в файл output.txt сумму этих чисел(гарантируется, что эта сумма помещается в int).
- **Сортировка** В файле input.txt лежат числа (сначала идёт число  $n$  – количество чисел, а затем идут  $n$  чисел). Вам нужно отсортировать эти числа (используйте qsort) и записать их в файл output.txt. Протестируйте вашу программу на файле numbers.txt.

## Посимвольное считывание из файла

Посимвольное считывание из файла с помощью функции `fgetc` позволяет просто решать некоторые задачи. Пример программы, которая находит количество цифр в файле:

```
FILE * f = fopen("input.txt", "r");
int c, number_of_digits = 0;

while ((c = fgetc(f)) != EOF)
{
    if (c >= '0' && c <= '9')
        number_of_digits += 1;
}
fclose(f);
```

- Написать программу, которая находит количество символов и строк в файле. Протестировать работу программы на книгах из папки `books/`.
- Написать программу, которая находит количество слов в файле. Слово это любая последовательность символов, разделённая пробелом, символом табуляции (`'\t'`) или символом перевода строки (`'\n'`). Протестировать работу программы на книгах из папки `books/`.
- Изменить программу из предыдущей задачи так, чтобы она принимала название файла через аргумент командной строки. Протестировать работу программы на книгах из папки `books/`.