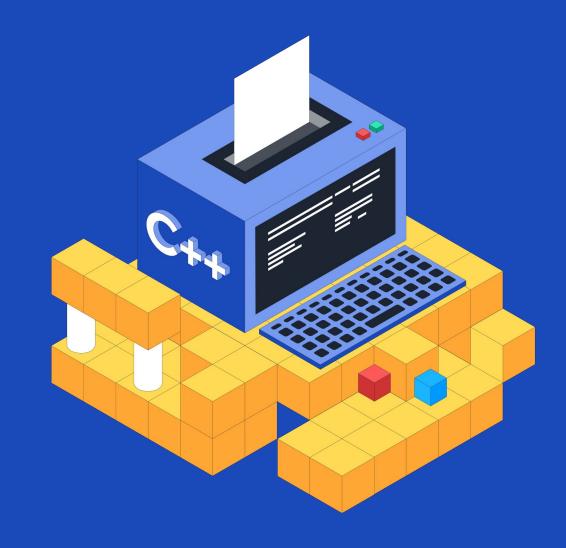




ПРОГРАММИРОВАНИЕ НА ЯЗЫКЕ C++

Материалы подготовлены отделом методической разработки

Средний уровень









Быстрая сортировка









Быстрая сортировка

Быстрая сортировка основывается на выборе опорного элемента (например, находящегося в середине массива) и перестановке элементов меньше опорного в левую часть массива, а элементов больше опорного - в правую часть массива.

Этот алгоритм относится к алгоритмам "Разделяй и властвуй".







Преимущества и недостатки

Быстрая сортировка отличается от других алгоритмов сортировки высокой скоростью выполнения. При этом реализация алгоритма остается довольно простой, хотя и осуществляется с использованием рекурсии.

Из основных недостатков - резкое увеличение времени выполнения при неудачных входных данных и неустойчивость.



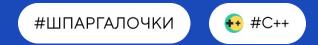




Реализация

Для реализации быстрой сортировки потребуются следующие шаги:

- 1. Создаем функцию, принимающую сортируемый массив, а также индексы первого и последнего элемента (**first** и **last**).
- 2. Первый и последний индексы сохраняются в дополнительные переменные (например, **f** и **l**).
- з. Вычисляем опорный элемент (mid).
- 4. Пока **f < I**: пока элемент под индексом **f** меньше опорного, увеличиваем f; пока элемент под индексом **I** больше опорного, уменьшаем I.
- 5. Если **f <= I**, меняем элементы с этими индексами местами, увеличиваем **f**, уменьшаем **l**.
- 6. Если **first < I**, рекурсивно вызываем функцию, в качестве **последнего индекса** передаем **I**. Если **f < last**, рекурсивно вызываем функцию, в качестве **первого индекса** передаем **f**.





Реализация

```
void qsort(int *mas, int first, int last) {
int f=first, l=last;
int mid=mas[(f+1) / 2];
do {
    while (mas[f]<mid) {
         f++;
         while (mas[l]>mid) {
             1--;
             if (f<=1) {
                  int count=mas[f];
                  mas[f]=mas[1];
                  mas[1]=count;
                  f++;
                  1--;
} while (f < 1);
if (first<1) qsort(mas, first, 1);
if (f<last) qsort(mas, f, last);
```





