1. cocktail\_sort**:**

* Сортирует массив книг по возрастанию цены методом перемешивания (**Shaker sort**).
* Этот алгоритм проходит массив в обе стороны:
  + Сначала слева направо, перемещая самый большой элемент вправо.
  + Затем справа налево, перемещая самый маленький элемент влево.
* **Сортировка перемешиванием**, или Шейкерная сортировка, или двунаправленная ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *Cocktail sort*) — разновидность [пузырьковой сортировки](https://ru.wikipedia.org/wiki/%D0%A1%D0%BE%D1%80%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%BA%D0%B0_%D0%BF%D1%83%D0%B7%D1%8B%D1%80%D1%8C%D0%BA%D0%BE%D0%BC). Анализируя метод пузырьковой сортировки, можно отметить два обстоятельства.
  + ***Во-первых***, если при движении по части массива перестановки не происходят, то эта часть массива уже отсортирована и, следовательно, её можно исключить из рассмотрения.
  + ***Во-вторых***, при движении от конца массива к началу минимальный элемент «всплывает» на первую позицию, а максимальный элемент сдвигается только на одну позицию вправо.

1. selection\_sort():  
   Реализует **сортировку выбором**:
   * Находит книгу с максимальным количеством страниц.
   * Меняет её местами с текущей позицией.
   * Сортирует книги по количеству страниц (по убыванию).
2. quick sort

Этот метод реализует итеративную версию **быстрой сортировки** (quick sort), чтобы избежать переполнения стека при глубокой рекурсии.

Процесс быстрой сортировки состоит из:

* Выбора опорного элемента (в данном случае — это последний элемент в подсписке).
* Разбиения списка на две части: одна с элементами больше опорного, другая с меньшими.
* Итеративного применения этого процесса для двух частей массива с использованием стека, вместо рекурсивных вызовов.