

19. Реализация быстрой сортировки без использования рекурсии

Описание алгоритма

В классической версии быстрой сортировки массив рекурсивно делится на две части: одна из которых содержит элементы меньшие опорного, а другая — больше. После этого две части массива сортируются отдельно, и процесс повторяется для каждой из частей.

Однако, используя стек, можно заменить рекурсию на итерацию. Алгоритм будет работать следующим образом:

1. Выбирается опорный элемент (pivot).
2. Массив делится на две части относительно опорного элемента.
3. Вместо рекурсии на каждом шаге алгоритм добавляет интервалы для сортировки в стек.
4. Пока в стеке есть интервалы, они обрабатываются по очереди.

Пример реализации быстрой сортировки без рекурсии

Пример кода на Python, который реализует быструю сортировку без рекурсии:

Листинг 1: Реализация быстрой сортировки без рекурсии

```
def quicksort_iterative(arr):
    stack = []
    stack.append((0, len(arr) - 1))

    while stack:
        low, high = stack.pop()

        if low < high:
            pivot_index = partition(arr, low, high)
            stack.append((low, pivot_index - 1))
            stack.append((pivot_index + 1, high))

    def partition(arr, low, high):
        pivot = arr[high]
        i = low - 1
        for j in range(low, high):
            if arr[j] <= pivot:
                i += 1
                arr[i], arr[j] = arr[j], arr[i]
        arr[i + 1], arr[high] = arr[high], arr[i + 1]
        return i + 1

#
arr = [10, 80, 30, 90, 40, 50, 70]
quicksort_iterative(arr)
```

```
print("                ~                :", arr)
```