

# ОЧЕРЕДИ И СОРТИРОВКА СЛИЯНИЕМ

КУРС

АЛГОРИТМЫ  
И СТРУКТУРЫ  
ДАННЫХ

СПИКЕР

Немков Максим Юрьевич

# Содержание темы

→ стек вызовов

→ Разделяй и властвуй

→ Быстрая сортировка

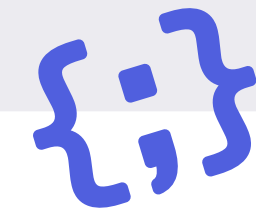
# Очередь



это структура данных, работающая по принципу "первый пришёл — первый вышел" (FIFO). Элементы добавляются в конец очереди и удаляются из её начала



## Сортировка слиянием



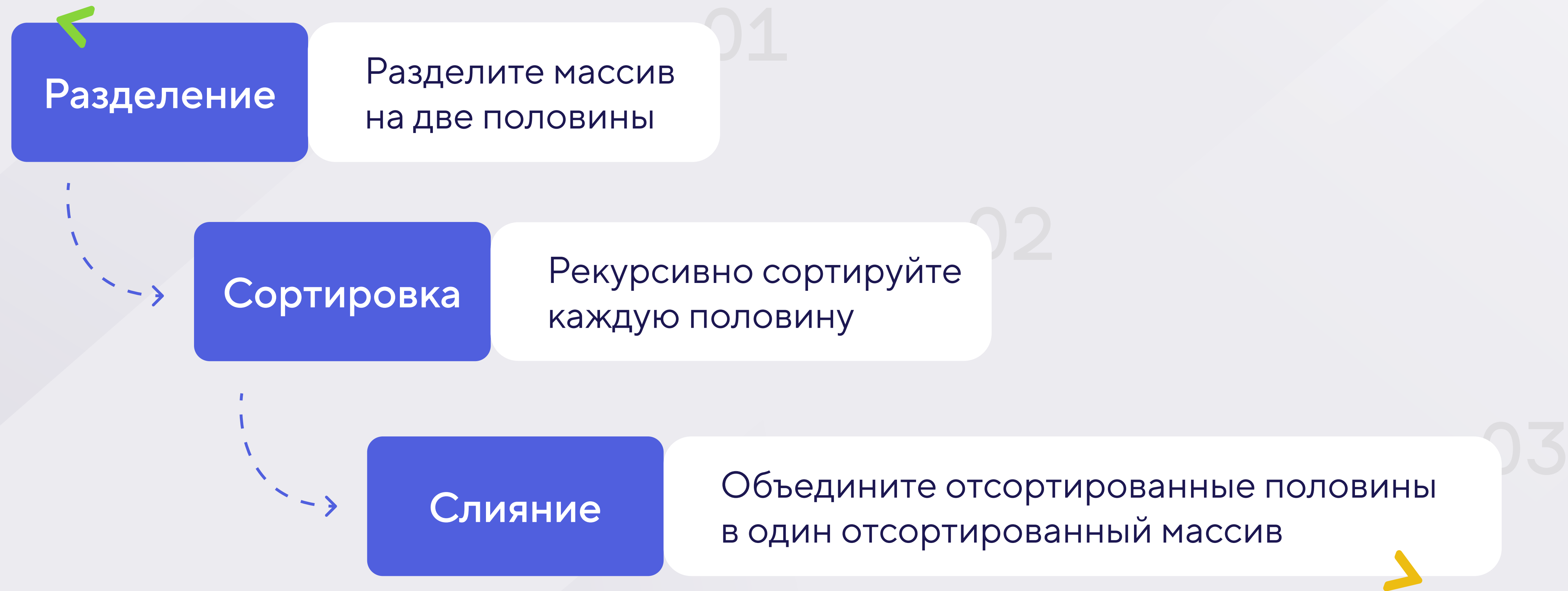
это алгоритм сортировки, использующий метод "разделяй и властвуй". Он разбивает массив на две части, рекурсивно сортирует каждую часть и затем объединяет отсортированные части в один массив






# Сортировка слиянием

Алгоритм работы сортировки слиянием





# Сортировка слиянием

Код сортировки слиянием



```
def merge_sort(arr):  
    if len(arr) > 1:  
        mid = len(arr) // 2  
        left_half = arr[:mid]  
        right_half = arr[mid:]  
  
        merge_sort(left_half)  
        merge_sort(right_half)  
  
    i = j = k = 0
```



```
# Слияние двух половин  
while i < len(left_half) and j <  
    len(right_half):  
    if left_half[i] < right_half[j]:  
        arr[k] = left_half[i]  
        i += 1  
    else:  
        arr[k] = right_half[j]  
        j += 1  
    k += 1
```



# Сортировка слиянием


Код сортировки слиянием



```
# Проверка остатков
```

```
    while i < len(left_half):  
        arr[k] = left_half[i]  
        i += 1  
        k += 1
```

```
    while j < len(right_half):  
        arr[k] = right_half[j]  
        j += 1  
        k += 1
```



```
# Пример использования:
```

```
arr = [38, 27, 43, 3, 9, 82, 10]  
merge_sort(arr)  
print("Отсортированный массив:", arr)
```

# ПОДВЕДЕМ ИТОГИ



Изучили, как работают очереди



Создали сортировку слиянием