

# Контрольные вопросы по алгоритмам сортировки

- 1) Количество сравнений в функции `void make_heap(int *a, int n);`
- 2) Количество сравнений в функции `int split_by_pivot(int *a, int n, int pivot_pos);`
- 3) Минимальное количество обменов при быстрой сортировке.
- 4) Количество сравнений при сортировке массива `[1,2,3,4,5,6,7,8,9,0]` выбором.
- 5) Количество сравнений при сортировке массива `[1,2,3,4,5,6,7,8,9,0]` вставками.
  
- 6) Опишите алгоритм пирамидальной сортировки.
- 7) Опишите алгоритм сортировки вставками.
- 8) Опишите алгоритм сортировки выбором.
- 9) Расскажите любой из алгоритмов `int partition(int *a, int n, int pivot_pos);`.
- 10) Опишите алгоритм слияния отсортированных массивов.
  
- 11) напишите пример `killer-sequence` длиной 16 для быстрой сортировки с алгоритмом выбора пивота: первый элемент.
- 12) Напишите пример последовательности длиной 16, максимизирующий время работы сортировки пузырьком.
- 13) Напишите последовательность, для которой время работы пирамидальной сортировки  $O(N)$ .  $N=10$ .
- 14) Напишите пример последовательности длиной 10, на котором достигается минимальное время работы quicksort с алгоритмом выбора пивота «первый элемент».
- 15) Приведите пример 2х отсортированных последовательностей длины 10, для которых количество сравнений при слиянии максимально.
  
- 16) Перечислите достоинства и недостатки алгоритма quicksort.
- 17) Предложите оптимальный алгоритм сортировки для типа данных с высокой стоимостью операции сравнения.
- 18) Предложите оптимальный алгоритм, для сортировки типа данных с высокой стоимостью операции swap.
- 19) В чём разница между методами восходящей и нисходящей сортировки слиянием.
- 20) Перечислите достоинства и недостатки алгоритма heapsort.
- 21) Предложите алгоритм слияния нескольких отсортированных списков ( $k > 2$ ).
  
- 22) Какова сложность алгоритма нахождения  $k$ -й порядковой статистики с алгоритмом выбора опорного элемента «медиана трёх» в худшем случае?
- 23) Какова сложность алгоритма нахождения  $k$ -й порядковой статистики с алгоритмом выбора опорного элемента «медиана медиан» в худшем случае?
- 24) Какова сложность алгоритма нахождения  $k$ -й порядковой статистики с алгоритмом выбора опорного элемента «случайный элемент» в худшем случае?

- 25) В чём смысл оптимизации концевой рекурсии?
- 26) Какое минимальное количество сравнений достаточно для сортировки 5 чисел?
- 27) Можно ли упорядочить 10 произвольных чисел, проведя 12 сравнений?
- 28) Как произвольный алгоритм сортировки сделать стабильным?
- 29) Поясните свойство стабильности алгоритма сортировки.
- 30) Поясните свойство локальности алгоритма сортировки.
  
- 31) Сколько двухпутевых слияний достаточно, чтобы отсортировать массив из 14 элементов?
- 32) Опишите алгоритм поразрядной сортировки от младших разрядов к старшим.
- 33) Опишите алгоритм поразрядной сортировки от старших разрядов к младшим.
- 34) Как реализовать алгоритм сортировки, работающий не медленее quicksort в среднем и с асимптотикой  $O(n \cdot \log(n))$  в худшем случае?
  
- 35) Предложите методы оптимизации алгоритма нисходящей сортировки слиянием.
- 36) Какому из методов сортировки соответствует наибольшее количество cache-miss в худшем случае при размере массива  $10^6$ : bubblesort, heapsort, quicksort?
- 37) в чём отличие алгоритмов binary quicksort и quicksort с выбором опорного элемента “середина”?
- 38) Какой метод лучше всего подходит для сортировки строковых ключей разной длины? Почему?
- 39) Какой метод лучше всего подходит для сортировки 6-значных десятичных чисел?
- 40) Постройте последовательность длины 10, приводящую к максимизации времени работы quicksort с алгоритмом выбора опорного элемента “середина”.
  
- 41) Предложите алгоритм частичной сортировки первых  $K$  элементов из массива размером  $N$  с помощью слияния?
- 42) Предложите алгоритм частичной сортировки первых  $K$  элементов из массива размером  $N$  с помощью кучи?
- 43) Опишите алгоритм сортировки слиянием.
- 44) Опишите алгоритм сортировки подсчётом.
- 45) Оцените затраты времени и памяти в алгоритме сортировки подсчётом.