

27) Односторонний двоякий поиск на  
 примере определенной границы серий  
 Односторонний двоякий поиск  
 используется для поиска границы, где  
 происходит переход между двумя  
 состояниями (указываются один серий и  
 начинается другой).

Пример: Определение границы серий.

Есть отсортированный массив:  $\{0, 0, 0, 1, 1, 1\}$

Требуется найти элемент с которого  
 начинается серия единиц:

$$\text{arr} = \{0, 0, 0, 1, 1, 1\}$$

$$\text{Начиная } \text{mid}_1 = \lfloor \frac{0+5}{2} \rfloor = 2$$

$$\Rightarrow \{0, 0, 0, 1, 1, 1\} \rightarrow \{0, 1, 1, 1\} \text{ т.к. } \text{arr}[\text{mid}_1] = 0 \text{ в подмассиве}$$

$$\text{mid}_2 = \lfloor \frac{0+1}{2} \rfloor = 0, \text{ ~~т.к. } \text{arr}[\text{mid}_2] = 0~~$$

$$\Rightarrow \{0, 1, 1, 1\} \rightarrow \{1, 1, 1\}$$

$$\{0\} \text{ т.к. } \text{arr}[\text{mid}_2] = 1 \text{ серия начинается с } \text{arr}[\text{mid}_2]$$

$$\text{т.к. } \text{arr} = \{0\} \text{ не содержит единиц } \text{mid} = \text{mid}_1 + \text{mid}_2$$

Временная сложность  $O(\log n)$

Пространственная  $O(1)$  (выделение памяти)



2.2) Задача определения пересечения множеств.  
решается в наихудшем случае элементами  
массива  $A$  или  $B$  или более подробные  
данные.

2 альтернативных способа решения:

1) Хэш-таблица (сортированные списки)

Временная  
сложность

$O(|A| + |B|)$

Пространств.

$O(|A|)$

1) Создается хэш-таблица по первому  
массиву

2) Проверяется по второму массиву:  
если по элементу второго массива  
существует запись в хэш-таблице  
и она не равна 0, то этот  
элемент встречается в обоих массивах.

2) Двухный поиск (сортированные списки)

Временная

$O(|A| \log |B| +$

$+ |A| \log |A|$

$+ |B| \log |B|)$

Пространств.

$O(1)$

1) брать элементы из второго массива и  
двухным поиском искать его в  
первом массиве

2) Если найден, добавляем в массив  $result$ ,  
если нет, переходим к следующему.

Повторять с каждым элементом  
массива  $B$ .



23) Определение дубликатов в массиве за  $O(n \log n)$   
Чтобы найти дубликаты за  $O(n \log n)$ , можно использовать сортировку. Дубликаты отмечаются.

1) Сортировка массива за  $O(n \log n)$

2) Линейный проход, попарное сравнение.

Временная сложность:  $O(n \log n + n) \rightarrow O(n \log n)$

Пространственная:  $O(1)$ , если сортировка на месте.

24) Задача определения частоты встречающихся элементов в массиве.

1) Использование хэш-таблиц, где ключи это значения массива, а значения их частоты

Временная сложность:  $O(n)$

Пространственная:  $O(n)$

2) Сортировка массива и линейный проход как в билете 23.