

## Максимальная сумма на подотрезке

Будем решать такую задачу: в заданном массиве целых чисел найти подотрезок с максимальной суммой.

Договоримся, что ответом будут два числа: левая и правая границы (индексы) такого отрезка.

Вам надо будет написать несколько решений этой задачи. Для удобства сравнения ответов надо найти крайний слева отрезок минимальной длины.

Если такой подход вызывает трудности, пусть ответом будет только значение максимальной суммы на подотрезке.

Обозначим далее буквой  $N$  размер входного массива.

Например:

для массива 3 -1 -5 5 3 -6 ответ 3 4,

для массива 1 2 2 4 ответ 0 3.

Несколько вопросов (приведите примеры, если ответ ДА):

- Могут ли быть в ответе отрицательные числа?
- Могут ли быть в ответе положительные и отрицательные числа?
- Могут ли быть в ответе *только* отрицательные числа?
- Каково максимальное количество отрицательных чисел в ответе для массива длины  $N$ ?
- Пусть вам известен ответ (границы подотрезка  $[L, R]$ ). Можно что-нибудь сказать про знаки элементов на концах этого подотрезка? знаки частичных суммы на префиксе и суффиксе этого подотрезка?

Для проверки ваших решений каждый вариант алгоритма надо оформить в виде функции. Затем сделайте массив из нескольких массивов-тестов. Это небольшие тесты, которые проверяют корректность решения на небольших массивах разных типов. Проверьте ваше первое решение и убедитесь, что оно даёт верный ответ.

После написания очередного варианта решения убедитесь, что оно даёт верные ответы на всех тестах.

---

A. Напишите решение с тремя циклами `for` за  $O(N^3)$  (напишите, даже если знаете, как решать быстрее).

*Указание:* первый цикл перебирает всевозможные значения левой границы подотрезка, второй цикл перебирает для текущей левой границы всевозможные значения правой границы, третий цикл вычисляет сумму элементов на подотрезке.

B. Перепишите первое решение, используя функцию `combinations` из модуля `itertools` и встроенную функцию `sum` для вычисления суммы среза массива. В решении должен быть один цикл `for`, перебирающий то, что возвращает `combinations`. Сравните время работы первого и второго решений на массивах размером до 500 элементов.

C. Чтобы быстро считать сумму любого подотрезка массива  $A$ , создайте сначала массив с его *префиксными суммами*:

`pref[k]` — сумма первых  $k$  элементов массива  $A$ .

Массив `pref` имеет длину на 1 бóльшую, чем массив  $A$  и `pref[0] = 0`. Перепишите второе решение и сравните время работы третьего и второго решений.

*Указание:* сумма элементов массива на отрезке  $[L, R]$  вычисляется так: `pref[R] - pref[L - 1]`.

D. Пусть сейчас проанализирован префикс массива из первых  $k$  элементов и вы храните:

- $\text{ansL}$  и  $\text{ansR}$  — границы наилучшего из пока виденных отрезков;
- $L$  и  $R$  — границы текущего отрезка, который имеет шансы стать новым ответом.

Подумайте, как может измениться ответ при рассмотрении элемента  $A[k]$ . Какие есть варианты?

Здесь полезно вспомнить последний вопрос из списка.

Перепишите решение с двумя указателями  $L$  и  $R$  и использованием массива с префиксными суммами. Это решение за  $O(N)$ .

E. Посмотрите на последнее решение и научитесь обходиться без массива префиксных сумм. Вы получите таким образом решение за  $O(N)$  без дополнительной памяти.

Например, такое решение подойдёт для задачи, где надо получить максимальную сумму на подотрезке последовательности чисел неизвестной заранее длины.