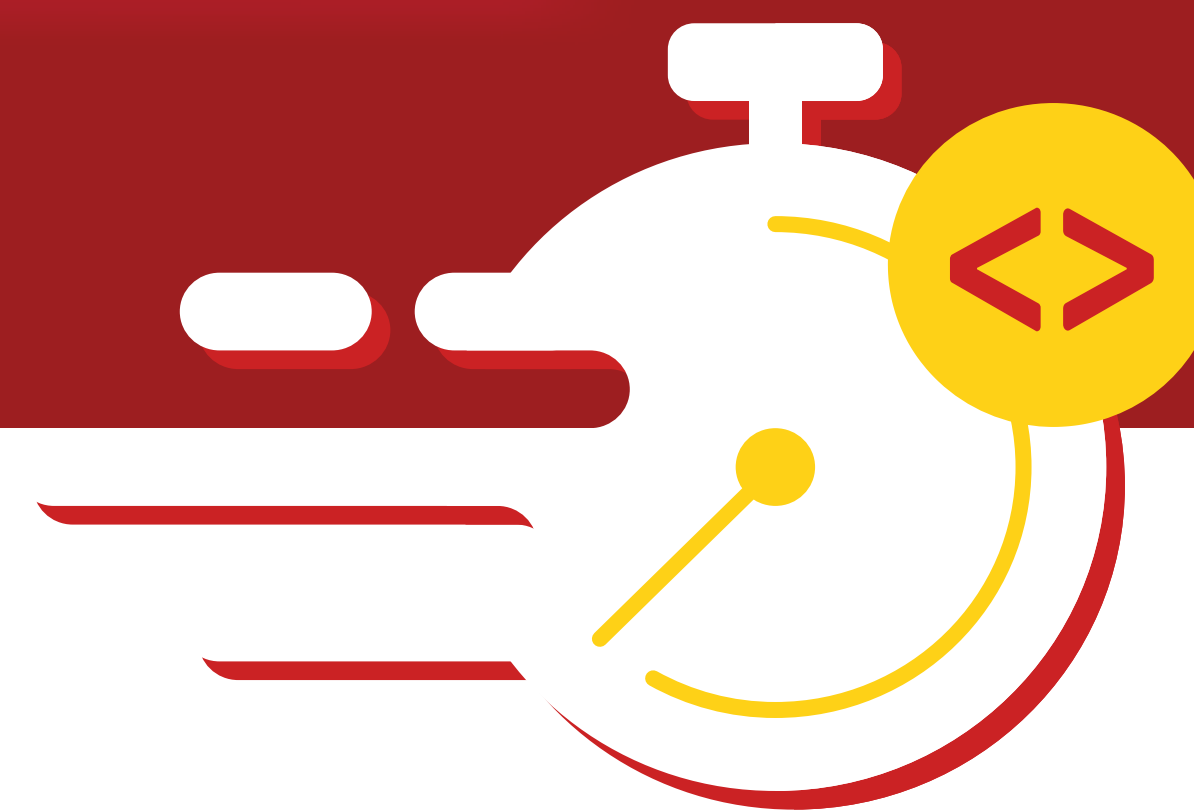
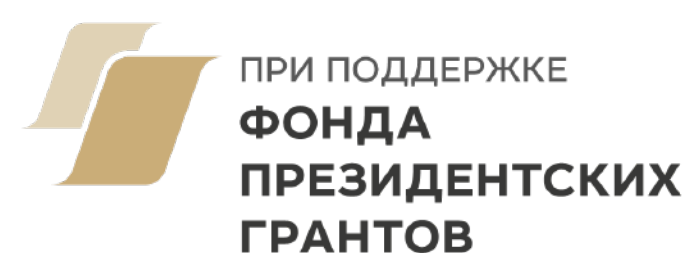


Линейные алгоритмы: суммы на подотрезке

Урок 1.2.2



Задача о сумме на подотрезке_

Пусть нам даны:

- Целое положительное число $n \leq 10^5$
- Массив из n целых чисел $a[1], a[2], \dots, a[n]$,
не превосходящих по модулю 10^9
- Целое положительное число $q \leq 10^5$
- q запросов вида $l, r, 1 \leq l \leq r \leq n$

Для каждого из запросов, требуется вывести
сумму $a[l] + a[l+1] + a[l+2] + \dots + a[r]$

Задача о сумме на подотрезке_

«Наивное» решение работает за $O(nq)$,
что слишком долго...

Метод частичных сумм_

Попробуем насчитать массив

$\text{partSum} = \text{partSum}[0..n]$, т.ч. $\text{partSum}[i] = a[1] + a[2] + \dots + a[i]$, $i = 0, 1, \dots, n$

- Для удобства считаем, что $\text{partSum}[0] = 0$

Метод частичных сумм_

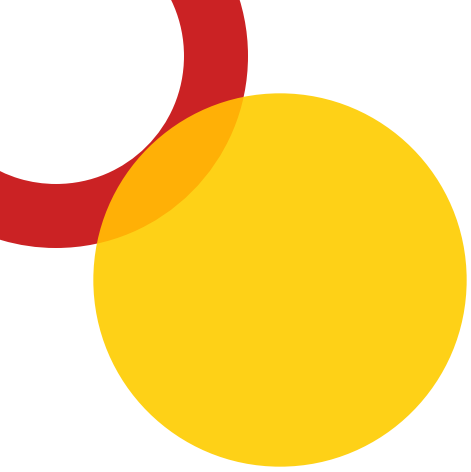
Если частичные суммы насчитаны,
то для любых l, r выполнено:

- $a[l] + a[l+1] + \dots + a[r] = \text{partSum}[r] - \text{partSum}[l-1]$

Следовательно, ответ на каждый из запросов
может быть получен за $O(1)$ —
если насчитан массив partSum

Метод частичных сумм_

- Насчитывать его наивным методом долго — общее число слагаемых будет $1 + 2 + \dots + n = n*(n+1)/2 = \Theta(n^2)$
- Заметим, что $\text{partSum}[i] = \text{partSum}[i-1] + a[i]$, $i = 1, 2, \dots, n$
- Следовательно, partSum насчитывается за $O(n)$!



```
1  #include <iostream>
2
3  using namespace std;
4
5  const int MAXN = 10000500;
6  int n;
7  int a[MAXN], partSum[MAXN];
8
9  int main() {
10     // ... заклинания для быстрого cin-a...
11
12     // предподсчет - O(n)
13     cin >> n;
14     partSum[0] = 0;
15     for (int i = 1; i <= n; ++i) {
16         cin >> a[i];
17         partSum[i] = partSum[i - 1] + a[i];
18     }
19
20     // обработка запросов - O(q)
21     int q;
22     cin >> q;
23     for (int query = 0; query < q; ++query) {
24         int l, r;
25         cin >> l >> r;
26         cout << partSum[r] - partSum[l - 1] << "\n";
27     }
28     return 0;
29 }
```

Линейные алгоритмы: итого_

- Познакомились с линейными алгоритмами, с тем, как их реализовывать и какие ошибки могут при этом возникать
- Изучили метод частичных сумм