

38. Вычисление ассоциативной функции в подмассивах фиксированной длины k за $O(n)$

Определение задачи

Пусть дан массив чисел $A = [a_1, a_2, \dots, a_n]$, и требуется вычислить ассоциативную функцию f в каждом подмассиве длины k , т.е. для каждого подмассива $A[i, i+k-1]$, где $1 \leq i \leq n-k+1$, необходимо вычислить $f(A[i, i+k-1])$.

Ассоциативная функция — это функция, для которой выполняется свойство ассоциативности, например:

$$f(x, z) = f(f(x, y), z)$$

Примером таких функций могут быть операции сложения, умножения, логические операции и другие.

Решение задачи за $O(n)$

Для вычисления ассоциативной функции за $O(n)$ можно использовать технику скользящего окна (два указателя), которая позволяет вычислять результат для каждого подмассива фиксированной длины k с использованием результатов предыдущего подмассива.

Шаги алгоритма

1. ****Вычисление значения функции для первого подмассива:**** Для первого подмассива $A[1, k]$ вычисляем ассоциативную функцию $f(A[1, k])$ напрямую, выполняя все операции на k элементах.
2. ****Вычисление для следующего подмассива:**** Для каждого последующего подмассива $A[i, i+k-1]$ используется результат для предыдущего подмассива $A[i-1, i+k-2]$. Мы добавляем новый элемент a_{i+k-1} и удаляем старый элемент a_{i-1} , что позволяет нам эффективно обновить результат ассоциативной функции:

$$f(A[i, i+k-1]) = f(f(A[i-1, i+k-2]), a_{i+k-1})$$

В случае операций, таких как сложение или умножение, этот шаг можно выполнить за постоянное время $O(1)$.

3. ****Повторение шага 2 для всех подмножеств массива:**** Алгоритм повторяет шаг 2 для каждого подмассива, пока не будет обработан весь массив. Таким образом, мы проходим по массиву один раз, обновляя значение функции для каждого подмассива.

Анализ сложности

Процесс вычисления ассоциативной функции с использованием скользящего окна позволяет нам обновлять результат для каждого нового подмассива за $O(1)$ времени. Поскольку мы обрабатываем каждый элемент массива ровно один раз, общая сложность алгоритма составляет $O(n)$, где n — количество элементов в массиве.

Таким образом, алгоритм эффективно решает задачу вычисления ассоциативной функции для подмассивов фиксированной длины k за время $O(n)$, что является оптимальным для данной задачи.