

Восстановление массива

Автор задачи: Софья Тарасова. Разработчик задачи: Михаил Грибакин

Удобно представить условия задачи в виде графа.

Каждому элементу массива a_i сопоставим вершину i . Каждому утверждению $a_x + a_y = s$ сопоставим ребро между вершинами x и y с весом s .

Рассмотрим одну компоненту связности этого графа. Пусть значение некоторой стартовой вершины равно неизвестному числу t .

Тогда по ребру (v, u) с весом s выполняется $a_u = s - a_v$.

Следовательно, значение каждой вершины компоненты можно представить в виде $a_v = t + shift_v$ или $a_v = -t + shift_v$.

Знак перед t меняется при каждом переходе по ребру. Его удобно хранить при помощи двух цветов вершин.

Во время DFS будем хранить:

- $color_v$ — знак перед неизвестным t ;
- $shift_v$ — свободную часть выражения для a_v .

Для стартовой вершины положим $a_{start} = t$, то есть $shift_{start} = 0$.

Если по ребру с весом s мы переходим из вершины v в ещё не посещённую вершину u , то $color_u = 1 - color_v$ и $shift_u = s - shift_v$.

Теперь рассмотрим ребро, которое ведёт в уже посещённую вершину.

Если цвета концов ребра различны, коэффициенты при t имеют разные знаки и взаимно уничтожаются. Такое ребро не позволяет определить t .

Если же цвета концов совпадают, то коэффициенты при t одинаковы. Например, если $a_v = t + shift_v$ и $a_u = t + shift_u$, то из условия $a_v + a_u = s$ получаем $2t + shift_v + shift_u = s$, откуда $t = \frac{s - shift_v - shift_u}{2}$.

Аналогично обрабатывается случай, когда обе вершины имеют коэффициент $-t$.

Таким образом, если в компоненте существует ребро между вершинами одного цвета, значение t определяется однозначно. После этого однозначно определяются значения всех вершин этой компоненты.

Если такого ребра нет, компонента является двудольной. Тогда t остаётся свободным параметром, поэтому ни одно значение в этой компоненте нельзя восстановить однозначно.

Иными словами:

- если компонента графа двудольная, ни один её элемент не определяется однозначно;
- если компонента не двудольная, все её элементы определяются однозначно.

После обхода каждой компоненты, если значение t удалось определить, восстанавливаем все её элементы по формулам $a_v = t + shift_v$ или $a_v = -t + shift_v$.

Для хранения графа достаточно в каждом ребре хранить номер соседней вершины и сумму s .

Каждое ребро и каждая вершина обрабатываются константное число раз, поэтому обход графа работает за $\mathcal{O}(n + m)$.

Если восстановленные индексы дополнительно сортировать перед выводом, итоговая асимптотика будет $\mathcal{O}(n + m + k \log k)$, где k — количество однозначно восстановленных элементов.

Можно также избежать сортировки и после обработки всех компонент пройти по индексам от 1 до n , выводя только восстановленные элементы. Тогда итоговая асимптотика останется $\mathcal{O}(n + m)$.

Так как промежуточные значения $shift_v$ могут достигать величин порядка $m \cdot s$, для всех вычислений значений массива необходимо использовать 64-битный целый тип, например `long long`.