Разбор задачи «Свежий взгляд»

В этой задаче достаточно было просто аккуратно вычислять уверенность Прохора после каждого высказывания Кеши: если уверенность Прохора и соображение Кеши имеют один знак, следовало прибавить к уверенности Прохора удвоенный вес сооображения. В противном случае (когда уверенность и соображение имеют разные знаки), следовало прибавлять вес соображения однократно. Возможные ошибки, в частности, могли быть связаны с некорректной обработкой нулевой уверенности: в этом случае также нужно было прибавлять вес соображения, взятый однократно.

Разбор задачи «Пицца»

«Классический» подход к решению этой задачи состоит в использовании техники двух указателей.

Сначала установим оба указателя на первый сектор и будем, двигая второй указатель, накапливать сумму углов s_1 до тех пор, пока она не станет равной или превысит 180^o. Тогда сумма оставшихся углов $s_2 = 360^o - s_1$. Найдем разность между этими величинами: $\Delta s = |s_1 - s_2| = 2 \cdot s_1 - 360^o$. Это первый кандидат на наилучшую разность. Сохраним позиции указателей как потенциальный ответ и значение Δs для дальнейших сравнений.

Далее будет делать следующее: вычтем величину первого сектора из суммы s_1, сместив таким образом первый указатель, после чего будем перемещать (если это необходимо) второй указатель, пока значение суммы s_1 вновь не окажется равным или большим 180^o. Когда это произойдет, вновь вычислим $|s_1 - s_2|$ и сравним его с предыдущим значением. Если оно оказалось меньше, то следует обновить сохраненные позиции указателей и значение Δs.

На следующем шаге вычтем из s_1 величину второго сектора и (при необходимости) добавим в s_1 следующие сектора, после чего снова выполним сравнение. Продолжая действовать таким образом, мы последовательно поместим первый указатель в каждый из секторов и построим сумму, наиболее близкую сверху к 180^o. Это позволяет нам перебрать все возможные разности, которые могут претендовать на звание минимальной, и найти ответ.

Разбор задачи «Долгий разговор»

Вероятно, самое сложное в этой задаче было аккуратно разобрать правила, по которым Иннокентий и Прохор выбирают очередную тему.

Поймем, как формируется последовательность обсуждаемых тем. От темы $t_1 = T_1 = 1$ происходит переход к любой другой теме, поскольку следующая тема не должна совпадать с текущей, а никаких иных тем до t_1 не обсуждалось (T_1 здесь обозначение числа 1).

Допустим, мы перешли к теме $t_2 = T_2 > 1$. Тогда следующей темой может быть либо T_1 (как наибольшая из обсуждавшихся ранее и не равная текущей), либо какая-то тема с $T_3 > T_2$. Если мы выберем T_1, то далее мы сможем вновь выбирать любую тему, кроме T_1, в том числе и T_2. Если же мы выберем некоторую тему T_3, то после нее мы сможем возвращаться к обсуждавшейся ранее T_2, но не к T_1. Вернуться к T_1 мы сможем только из T_2.

Продолжая рассмотрение дальше, мы можем заметить, что, если мы перешли от темы с номером T_j к теме с номером T_{j+1}, и при этом $T_j < T_{j+1}$, то возвратиться из темы T_{j+1} мы можем только в T_j. Действительно, эта тема не является текущей (и подходит по этому критерию). Второй критерий также удовлетворяется: нас интересует тема с наибольшим номером, с момента предыдущего обсуждения которой не обсуждались темы с меньшим номером. В частности, такая тема должна иметь номер, меньший T_{j+1}. Среди всех таких (обсуждавшихся ранее) тем правильным выбором будет именно T_j. Если есть некая тема с номером $T_{j'}$, таким, что $T_j < T_{j'} < T_{j+1}$, то это значит, что после $T_{j'}$ обсуждалась T_j (т.е. тема с меньшим номером), и потому $T_{j'}$ не подходит.

Де-факто это означает, что «возвращаться» к темам товарищи будут тем же путем, которым они от них уходили. Таким образом, если у нас имеется фрагмент разговора, в котором номера возрастают, то к любому элементу этого фрагмента мы можем вернуться, пройдя все следовавшие за ним номера в обратном порядке. Т.е. весь разговор должен выглядеть как чередование возрастающих и убывающих последовательностей, причем убывающая последовательность является «зеркальным отражением» части предшествующей ей возрастающей. Возможно, в этом описании уже можно «узнать» такую структуру данных как стек.

После этого становится принципиально понятно, как можно действовать (мы не утверждаем, что это единственно верная последовательность действий).

Если очередная тема имеет больший номер, нежели текущая, то считаем, что эта тема является непосредственным продолжением разговора, и учитываем один переход.

Если очередная тема имеет меньший номер, нежели текущая, то нам придется возвращаться: либо до этой темы, если такое возможно (и никаких переходов от меньшего номера к большему не произойдет), либо до первой темы, имеющей меньший номер — и тогда нам потребуется один переход.

Номера тем, конечно, удобно хранить в стеке (если в используемом Вами языке нет такой структуры данных, можно использовать массив достаточной длины и индекс–указатель на «вершину стека»). Тогда номер, больший номера на вершине, помещается в стек и сам становится вершиной, а номер, меньший номера на вершине, приводит к удалению части элементов из стека. Заметим, что мы никогда не будем удалять из стека больше элементов, чем мы туда поместили.

Разбор задачи «Полезные знания»

Видимо, самое простое в реализации решение выглядит следующим образом. Разобьем плоскость на квадратные блоки размером 700×700. Поскольку диагональ такого квадрата $700 \cdot \sqrt{2} < 1000$, в одном блоке не может быть двух станций метро. Так как $3 \cdot 700 = 2100 > 2000$, все станции метро, находящиеся в блоках, отстоящих на 4 и больше по горизонтали или вертикали от блока, в котором находится достопримечательность, расположены на расстоянии более 2000 и нас не интересуют. Таким образом, если достопримечательность попала в блок с координатами (x, y), достаточно проверить станции метро, расположенные в блоках $([x-3; x+3], [y-3; y+3])$, всего 49 штук.

Кроме того, возможно решение этой задачи упорядочиванием станций метро по одной из координат с последующим сканированием полосой ширины 4000.

Наконец, возможно решение этой задача алгоритмом «разделяй-и-властвуй», во многом аналогичное стандартному алгоритму для задачи о паре ближайших точек (см., например, `http://e-maxx.ru/algo/nearest_points`).

Разбор задачи «Точка зрения»

Самое простое решение этой задачи, видимо, состоит в следующем. Будем рассматривать приходящие оценки по одной слева направо. Будем хранить все варианты разбиения последовательности оценок, которые нам еще могут пригодиться. На каждой итерации будем получать все новые возможные разбиения, пытаясь добавить рассматриваемый элемент в каждую подпоследовательность каждого разбиения.

Определим для каждого разбиения a пару чисел (a_1, a_2) — последние числа неубывающей и невозрастающей подпоследовательностей разбиения. Тогда, если для двух разбиений a и b одновременно верно $a_1 \leqslant b_1$ и $a_2 \geqslant b_2$, то любое правильное продолжение разбиения b будет правильным продолжением разбиения a, и мы можем забыть про b. Заметим, что одним из (a_1, a_2) является только что рассмотренное число. Таким образом, из любых трех разбиений одно будет лишним, а, стало быть, их всегда не более двух.

Наивная реализация подхода, описанного выше, потребует $O\left(n^2\right)$ времени из-за необходимости вывести (и копировать на каждой итерации) пример разбиения. Существует как минимум две возможности, как получить реализацию за $O(n)$:

- использовать технику динамического программирования и стандартную схему восстановления ответа;
- использовать три массива, в одном из которых записывается общая часть для двух разбиений, а в двух других — продолжения, соответствующие каждому из разбиений (или, аналогично, односвязные списки, направленные в обратную сторону).

Из других возможных решений отметим подход, основанный на максимумах и минимумах. Краткий план: начнем со всего набора и будем строить разбиения для префикса и суффикса. Если максимальный элемент является первым элементом подотрезка набора, никогда не будет плохо отправить его в невозрастающую последовательность, если, конечно, там еще нет элементов, больших него. Аналогично для максимального, расположенного в конце подотрезка, и для минимального на границе подотрезка. А если максимальный и минимальный элементы не лежат на границе подотрезка, то они не могут быть в разных подпоследовательностях, и мы опять же можем распределить их. Однако данное решение весьма трудоемко в реализации, в особенности со сложностью $O(n)$.

Кроме того, по-видимому, возможно решение этой задачи разбором случаев после реализации следующей идеи: давайте найдем наибольшую невозрастающую и неубывающую подпоследовательности, их суммарная длина будет как минимум n. Дальше нужно разобраться, что делать с пересечением и невыбранными элементами.

Относительно частичных решений в этой задаче стоит отметить, что перебор с возвратом получает примерно 90 баллов, поскольку работает за $O\left(n^2\right)$.