Разбор задачи «Экономическая грамотность»

В данной задаче надо было сравнить три числа: $S \cdot 0.03$, $\lfloor \frac{S}{50} \rfloor \cdot 2$ и $\lfloor \frac{S}{100} \rfloor \cdot 5$ и, в зависимости от результата, вывести `MM`, `BB` или `RR`. Чтобы избежать проблем с вещественной арифметикой, можно было считать не сами эти значения, а значения, увеличенные в 100 раз, тогда каждое из трёх чисел будет целое.

Асимптотика решения $O(1)$.

Разбор задачи «Наша Таня громко плачет»

Если $k = 1$, то ответ на задачу, очевидно, $(n-1) \cdot A$, в противном случае будем жадно уменьшать количество файлов жадно.

В каждый момент времени мы можем находиться в одной из трёх ситуаций:

1. Если $n < k$, то нам ничего не остается, кроме как $n - 1$ раз уменьшить число за A, мы можем сделать это за $O(1)$ формулой.

2. Если $n > k$ и n не кратно k, то нам ничего не остается, кроме как $(n \, mod \, k)$ уменьшить число за A, эту часть тоже можно сделать за $O(1)$ формулой.

3. Если n кратно k, то нам всегда выгодно сделать переход к $\frac{n}{k}$ за $min(B, (n - \frac{n}{k}) \cdot A)$. Если $B < (n - \frac{n}{k}) \cdot A$ корректность очевидна. Иначе предположим мы не сделали переход к $\frac{n}{k}$ сейчас, а сделали его на отрезке $(\frac{n}{k}; n)$ из числа $n - i \cdot k$, тогда мы заплатили $min(B, (n - i \cdot k - \frac{n}{k} + i) \cdot A) + i \cdot k \cdot A$. Это $(n - i \cdot k - \frac{n}{k} + i) \cdot A + i \cdot k \cdot A = (n - \frac{n}{k}) \cdot A - i \cdot (k - 1) \cdot A + i \cdot k \cdot A = (n - \frac{n}{k}) \cdot A + i \cdot A$ или $B + i \cdot k \cdot A$, что не выгоднее перехода сразу к $\frac{n}{k}$ и последующему спуску к числу $\frac{n}{k} - i$. Данный переход тоже делается за $O(1)$.

Поскольку в каждой ситуации мы можем побывать не более $\log_k n$ раз асимптотика решения $O(\log_k n)$.

Разбор задачи «Новое — это хорошо забытое старое»

Рассмотрим 2 случая:

1. Если $n < k$, то надо приписать в конец исходной строки $k - n$ минимальных символов из s.

2. Если $n \geqslant k$, то надо суффикс строки, состоящий из первых k символов исходной строки, из самых больших символов заменить на самые минимальные символы, а следующий символ после этого суффикса увеличить до следующего в алфавите символа, присутствующего в исходной строке.

Асимптотика $O(n + k)$.

Разбор задачи «Умный обогреватель»

Заметим, что ограничения на t_{min} и t_{max} задают только последовательности работы обогревателя равные `00001`, `00000`, `11110`, `11111`.

1. `00000` на позициях $[i; i + 4]$ значит, что $t_{min} \leqslant \max\limits_{i \leqslant j \leqslant i+4} a_j$.

2. `00001` на позициях $[i; i + 4]$ значит, что $t_{min} \geqslant \max\limits_{i \leqslant j \leqslant i+4} a_j + 1$.

3. `11111` на позициях $[i; i + 4]$ значит, что $t_{max} \geqslant \min\limits_{i \leqslant j \leqslant i+4} a_j$.

4. `11110` на позициях $[i; i + 4]$ значит, что $t_{max} \leqslant \min\limits_{i \leqslant j \leqslant i+4} a_j - 1$.

В итоге, мы получаем набор ограничений на верхнюю и нижнюю границу для t_{min} и t_{max}. Поскольку гарантируется, что ответ существует, то есть всего два случая:

1. Отрезки пересекаются, тогда ответ — $t_{min} = t_{max} =$ любое число из пересечения.
2. Отрезки не пересекаются, тогда надо выбрать максимально возможное значение t_{min} и минимально возможное значение t_{max}.

Асимптотика $O(n)$.

Разбор задачи «Оптимизация акции»

Сначала решим задачу за $O(n^2 \log n)$. Это можно сделать при помощи динамики dp_i — минимальная стоимость разбиения префикса массива длины i. Пересчет $dp_0 = 0$, $dp_i = \min\limits_{j<i} dp_j + cost_{j+1,i}$, где $cost_{l,r}$ — сумма $(r - l + 1) - \lfloor \frac{r-l+1}{10} \rfloor$ максимумов на подотрезке $[l; r]$. В момент пересчета мы можем идти по j от $i-1$ до 0, поддерживая текущие элементы массива на подотрезке $[j+1; i]$ в $std :: multiset$ и аккуратно пересчитывая сумму $\lfloor \frac{r-l+1}{10} \rfloor$ минимумов.

Чтобы решить задачу быстрее, надо понять, что нам всегда выгодно брать отрезки либо длины 1, либо длины 10. Предположим мы взяли отрезок длины меньшей 10, тогда его стоимость не зависит от разбиения и его можно разбить на отрезки длины 1, предположим мы взяли отрезок длины x, $10 \leqslant x \leqslant 19$, тогда мы можем оставить из этого отрезка оптимальный подорезок длины 10, а все что по краям взять отрезками длины 1. Если длина отрезка равна 20, то, очевидно, не хуже взять его как два отрезка длины 10. В остальных случаях можно также предъявить разбиение на подотрезки длины 1 и 10, которое будет не дороже исходного.

Делать пересчет для отрезков длины 1 легко, чтобы делать пересчеты для отрезка длины 10, можно хранить элементы из отрезка $[i - 9; i]$ в любой структуре данных, умеющей брать минимум быстро. Например очередь с минимумом или $std :: multiset$.

Асимптотика $O(n)$.