

Тест на интеллект

Автор задачи и разработчик: Даниил Голов

Переформулируем условие без легенды. Необходимо составить из букв латинского алфавита строку длины n . Далее использованной в строке букве алфавита будет сопоставлена стоимость от 1 до количества уникальных букв так, чтобы суммарная стоимость всех букв строки была минимальна. Нужно найти строку максимальной возможной итоговой стоимости.

Для начала докажем, что для этого необходимо, чтобы количество вхождений любых двух букв в строку отличалось не более чем на 1. Обозначим за $\text{count}(c)$ количество вхождений буквы c в строку. Докажем от противного — пусть есть такие две буквы x и y , что $\text{count}(x) - \text{count}(y) \geq 2$. Тогда при сопоставлении стоимостей букве y будет назначена стоимость q , букве x — стоимость p , и $p < q$, так как если $p > q$, то

- буквы x и y дают вклад $\text{count}(x) \cdot p + \text{count}(y) \cdot q$ в стоимость строки;
- если стоимости этих букв поменять местами, то получится меньшее значение $\text{count}(x) \cdot q + \text{count}(y) \cdot p$.

А в таком случае, если заменить одну букву x на букву y , то стоимость, которую бы две эти буквы принесли в сумму, была бы $(\text{count}(x) - 1) \cdot p + (\text{count}(y) + 1) \cdot q$, что на $q - p$ больше. Значит, такая строка не является максимальной по стоимости, что и требовалось доказать.

Более того, нам «выгодно» брать больше различных букв. Действительно, если мы рассмотрим строку, в которой $a < 26$ различных букв, и заменим одну самую частую из них на еще неиспользованную, мы получим большую суммарную стоимость (как следует из предыдущего утверждения, машина назначает меньшие веса более часто встречающимся буквам).

То есть, нам нужно как можно больше разных букв, и при этом как можно более равные количества различных букв (отличающиеся не более чем на 1). Тогда для ответа, например, подойдет строка « $abcdef \dots zabcdef \dots zabc \dots$ » длины n . Если $n \leq 26$, можно вывести просто первые n букв латинского алфавита. Решение чисто конструктивное, работает за время $\mathcal{O}(n)$.