

Зашифрованное сообщение

Автор задачи: Владислав Власов, разработчик: Даниил Орешников

Для решения первых подзадач можно воспользоваться методом динамического программирования, $dp[i]$ — минимальное число слов, на которые указанным образом можно разбить префикс и суффикс длины i . Пересчет достаточно стандартный: для текущего i перебираем начало последнего слова $j < i$, после чего

1. проверяем, что $s[j : i] = s[|s| - i : |s| - j]$;
2. если они равны, обновляем $dp[i]$ через $dp[j]$.

Проверять строки на равенство можно перебором символов за время $\mathcal{O}(|s|)$, а можно с помощью хешей за $\mathcal{O}(1)$. Эти два решения проходили разный набор подгрупп.

При условии, что исходная строка состоит только из букв 'a' и 'b' решение не сильно отличается от полного, однако на этой подгруппе может быть проще доказать, что жадный алгоритм действительно приводит к оптимальному результату.

Действительно, решим задачу жадно: будем каждый раз брать минимально возможные по длине равные префикс и суффикс и отрезать их с двух сторон от строки. Докажем, что это приводит к оптимальному ответу.

Пусть это не так, тогда в очередной раз лучший ответ получается взятием большей по длине подстроки с концов строки s . В таком случае s представима как $s = t_1 + q_1 + t_1$ и $s = t_2 + q_2 + t_2$, где $|t_1| > |t_2|$. Но при таком представлении s несложно заметить, что t_2 является как префиксом, так и суффиксом t_1 , то есть является ее бордером. Минимальный бордер строки не превышает половины ее длины, поэтому $t_1 = t_2 + r + t_2$, и тогда s можно было представить как $t_2 + r + t_2 + q_2 + t_2 + r + t_2$, что является ее разбиением на еще большее число слов, что и требовалось доказать.

Таким образом, для полного решения достаточно было посчитать хеши префиксов строки s , после чего жадно набирать слова в ее разбиении, проверяя суффикс и префикс на равенство с помощью этих хешей. Общее время решения — $\mathcal{O}(|s|)$.