

Гараж

Автор задачи и разработчик: Павел Скобелин

Заметим, что можно сразу избавиться от паролей, которые являются подстроками других строк. Для этого переберем все возможные пары строк, и проверим, находится ли строка s_i в строке s_j ($i \neq j$) с помощью любого подходящего алгоритма (к примеру, алгоритм КМП или хеши). Также нужно быть аккуратными с одинаковыми строками.

Теперь давайте построим ориентированный граф на n вершинах. Длина ребра (i, j) будет равна минимальному количеству символов, которые необходимо дописать в конец s_i , чтобы полученная строка содержала как подстроку строку s_j (менее формально, нас интересует максимальное «наложение» s_j справа на s_i). Давайте научимся считать эту величину: так как никакие две строки теперь не содержатся друг в друге, найдем максимальный суффикс s_i , совпадающий с префиксом s_j такой же длины. Тогда несложно понять, что искомая величина — это разность длины s_j и длины максимального совпадающего суффикса s_i и префикса s_j . А максимальные совпадающие префикс и суффикс можно найти, взяв последнее значение префикс-функции от строки « $s_j\#s_i$ », или посчитав аналогичное значение с помощью хешей.

Теперь заметим, что ответ на задачу — это гамильтонов путь минимальной длины в приведенном выше графе, с учетом того, что стартовать i -й вершины «стоит» длину строки s_i . Это несложно показать: посмотрим на вхождения данных строк в ответ: они идут в каком-то порядке. Очевидно, что нет смысла включать одну строку в этот порядок более 1 раза, ведь на графе выполняется неравенство треугольника. А чтобы для данного порядка минимизировать ответ, необходимо максимизировать «наложение» соседних пар строк — именно это мы и максимизируем в искомом графе.

В итоге мы получаем решение за $\mathcal{O}(2^n \cdot n^2 + S \cdot n^2)$ на поиск гамильтонова пути минимальной стоимости, и удаление строк, входящих в другие.