

Домашнее задание 11. Сложные строковые алгоритмы
Автор: *Головко Денис*, Б05–225

Загадка 1.

Решение.

Преобразуем строку следующим образом: $s_1 \dots s_n \mapsto s_1 \# \dots \# s_n \$ s_n \# \dots \# s_1$. Это необходимо, чтобы одновременно учитывать как исходные, так и обратные суффиксы. За $O(n)$ построим суффиксный массив и посчитаем на нем массив LCP .

Будем идти по суффиксному массиву. Если рядом находятся суффиксы, принадлежащие как исходной, так и реверснутой строке, то мы нашли палиндром, то есть ответ для позиции можно будет обновить как максимум из ответа и LCP для этих суффиксов, при этом начальный ответ для позиции положим равный 1.

Асимптотика. $O(n)$

Загадка 2.

Решение.

Преобразуем строки в одну следующим образом: $s_1, \dots, s_n \mapsto s_1 \# \dots \# s_n$. На новой строке построим суффиксный массив и массив LCP .

Если s_i входит в другое слово s_j как подслово, то это можно определить по значению LCP для соответствующих подстрок. LCP для строки s_i и следующей за ней строки в суффиксном массиве будет $\geq |s_i|$. Корректность следует из лексикографического порядка.

Асимптотика. $O(\sum_{i=1}^n |s_i|)$

Загадка 3.

Решение.

Построим суффиксный автомат для объединенной строки $s_1 \# \dots \# s_n$. Суффиксный автомат позволяет быстро проверять вхождения подстроки в строку, а также находить длины наибольших общих суффиксов. Проходя по каждой из строк s_i , будем обновлять длину наибольшего суффикса, который является префиксом другой строки, при достижении каждой терминальной ноды суффиксного автомата.

Для ответа на запрос о наибольшем суффиксе, который является префиксом другой строки, будем обращаться к уже вычисленному двумерному массиву.

Асимптотика. $O(n^2 + \sum_{i=1}^n |s_i|)$

Загадка 4.

Решение.

Построим суффиксный автомат на строке s . Суффиксы строки s попарно несравнимы, если им можно сопоставить пути в суффа автомате, не пересекающиеся по вершинам.

Из всех завершающих состояний автомата проведем ребро в фиктивную вершину и раздвоим вершины на v_{in} и v_{out} . Далее воспользуемся алгоритмом Форда-Фалкерсона для нахождения максимального потока из начальной вершины в общую терминальную. Максимальный поток будет равен кол-ву вершинно-непересекающихся путей в автомате. Кроме того, максимальный поток ограничен сверху мощностью константного алфавита.

Асимптотика. $O(n)$