МФТИ, ФПМИ
Алгоритмы и структуры данных, осень 2021
Семинар №02. Бор, алгоритм Ахо—Корасик

1. По префикс-функции строки постройте её z-функцию.
2. Дана строка s. С помощью линейного предподсчёта научитесь отвечать на запросы вида "является ли подстрока $s_l \ldots s_r$ палиндромом?" за $O(1)$.
3. В множестве S лежат n чисел: $x_1, \ldots, x_n$. Для каждого из чисел $\{y_i\}_{i=1}^q$ найдите $\max\limits_j (y_i \oplus x_j)$. Проделайте то же для минимума. В предположении, что все числа — целые и лежат в отрезке $[0, 2^k - 1]$, отвечайте на каждый запрос за $O(k)$.
4. К множеству S поступают запросы двух видов: добавить строку в S; сообщить k-ю строку S в лексикографическом порядке (k — параметр запроса). Отвечайте на запрос за линейное время от длины строки. Как с помощью аналогичной техники отсортировать заданный список строк?
5. Как реализовать алгоритм Ахо—Корасик на динамически расширяющемся множестве строк? Используйте идею разложения n по степеням двойки.
6. Дана строка s с не более чем k знаками вопроса. Вхождением s в текст t назовём подстроку t, которая совпадает с s во всех символах, кроме вопросов. Найдите все вхождения s в t за время $O((|s| + |t|) \cdot k)$.
7. Рассмотрим алфавит из 4 букв и n строк в нём: $s_1, \ldots, s_n$, причём $|s_i| \leqslant m$ для всех i. Слово t назовём хорошим, если в нём можно выделить несколько подстрок, каждая из которых равна какому-нибудь s_i, и все символы при этом находятся хотя бы в одной из выделенных подстрок (то есть t покрыта словарным словами). Найдите число хороших строк длины k.
8. Задан фиксированный словарь, каждое слово имеет свой вес. Поступают запросы двух видов: "изменить вес i-го словарного слова" и "найти словарное слово с максимальным весом, которое входит в текст t как подстрока". Отвечайте на запросы первого типа за $O(\log^2 n)$, где n — суммарная длина слов словаря, а на запросы второго типа — за $O(|t| \log^2 n)$.

1. По префикс-функции постройте какую-нибудь подходящую строку (с такой префикс-функцией). По ней уже за линейное время постройте z-функцию.
2. Воспользуйтесь алгоритмом Манакера. А именно, для каждого i найдите самый длинный палиндром с центром в i. Это можно делать так же, как ищется z-функция.
3. Сложите $x_1, \ldots, x_n$ в бор, представив их битовыми строками одной длины. Для поиска ответа спускайтесь жадно от старших битов к младшим.
4. В каждой вершине бора храните параметр — количество терминальных вершин в поддереве.
5. Храните несколько боров, каждый из которых хранит информацию о 2^k строках (для некоторого k), и все такие k в структуре различны.
6. Сложите сплошные подстроки s в бор (разбив s по знакам вопроса). Чтобы проверить, входит ли s в t в данной позиции, нужно проверить, что в этой позиции входит последняя подстрока s, чуть левее — предпоследняя, и т.д. Можно проделать то же самое, используя только префикс-функцию.
7. Введите динамику по автомату Ахо—Корасика: храните вершину, число использованных символов, а также самый длинный непокрытый суффикс. При добавлении нового символа может появиться вхождение одного из словарных слов, то есть непокрытый суффикс может опустеть.
8. Постройте автомат Ахо—Корасик на словаре. Постройте дерево отрезков на эйлеровом обходе дерева суффиксных ссылок в этом автомате. Тогда нужно поддерживать множества, добавлять и удалять из них элементы, а также узнавать максимум. Решение — дерево отрезков мультисетов.