

## План лекции 4

# Словари. Деревья. Красно-черные деревья

### 4.1 Словари

Словарь — структура данных, имеющая операции

- INSERT;
- DELETE;
- FIND.

В ближайшее время мы изучим один из вариантов реализации словаря — дерево поиска; добиваться будем реализации операций за  $O(\log(\text{количества элементов}))$  операций над элементами. Затем мы изучим другой вариант — хеш-таблицы.

### 4.2 Деревья

#### 4.2.1 Определения

Граф, дерево, дерево с корнем.

**ПРОБЕЛ В КОНСПЕКТЕ.**

#### 4.2.2 Представление деревьев в компьютере

Будем говорить о деревьях с корнем.

Имеется много разновидностей структур данных, называемых деревьями. Соответственно, и набор операций над деревьями может быть разным. Поэтому *представление зависит от требуемых операций*.

Как представить двоичное дерево, ясно: элемент, соответствующий каждой вершине, содержит хранимые данные и два указателя: на левого и правого сына; если нужно — указатель на родителя. Для дерева с произвольной степенью вершин имеется несколько вариантов: например, список сыновей (полезно указать их количество, а указатель наверх хранить в каждом из них).

**Замечание 4.1.** Динамический массив вместо списка не подойдет — трудно вставлять!

### 4.3 Красно-черные деревья

Данные будут храниться в вершинах дерева, но обращать внимание мы будем только на ключ — то поле записи, по которому ведется поиск. Ключ должен быть элементом некоторого линейно упорядоченного множества. Даже если исходно данные не представлены в виде дерева, их может быть полезно представить в таком виде, если они упорядочены и над ними часто приходится выполнять указанные операции.

Свойство

«все элементы левого поддерева меньше либо равны ключу, ключ больше всех элементов правого поддерева» (4.1)

позволяет реализовать FIND за  $O(\text{высоты дерева})$  операций. Если дерево «идеально» (высота  $h$ , количество вершин  $2^{h+1} - 1$ ), имеем  $O(\log n)$  операций. Ниже мы изучим разновидности деревьев поиска, позволяющие поддерживать себя в «почти идеальном» виде, при этом ограничиваясь логарифмическим количеством операций с элементами при реализации операций INSERT и DELETE.

Определение красно-черного дерева, теорема о высоте, реализация операции INSERT — см. книгу Кормена, Лейзерсона и Ривеста.

**ПРОБЕЛ В КОНСПЕКТЕ.**

### 4.4 Очереди с приоритетами

Очередь с приоритетом — структура данных, имеющая операции

- INSERT;
- MAX (показывает элемент с максимальным ключом);
- EXTRACT-MAX (показывает и удаляет элемент с максимальным ключом).

Заметим, что красно-черные деревья реализуют очередь с приоритетом с логарифмической сложностью всех ее операций. В дальнейшем мы изучим и другую реализацию, в которой операции будут реализованы еще эффективнее.

## 4.5 Порядковая статистика для красно-черных деревьев

См. книгу Кормена, Лейзерсона и Ривеста.

**ПРОБЕЛ В КОНСПЕКТЕ.**