

N-арные деревья

Обобщением бинарных деревьев являются n-арные деревья, то есть деревья со степенью исхода n .

Характерная черта n-арных деревьев – узел содержит один ключ и n указателей.

Наиболее популярными деревьями этого типа среди программистов можно считать:

- деревья цифрового поиска,

- trie – деревья или нагруженные деревья,

- patricia – деревья,

- синтаксические деревья и другие.

Преимущества и недостатки n-арных деревьев

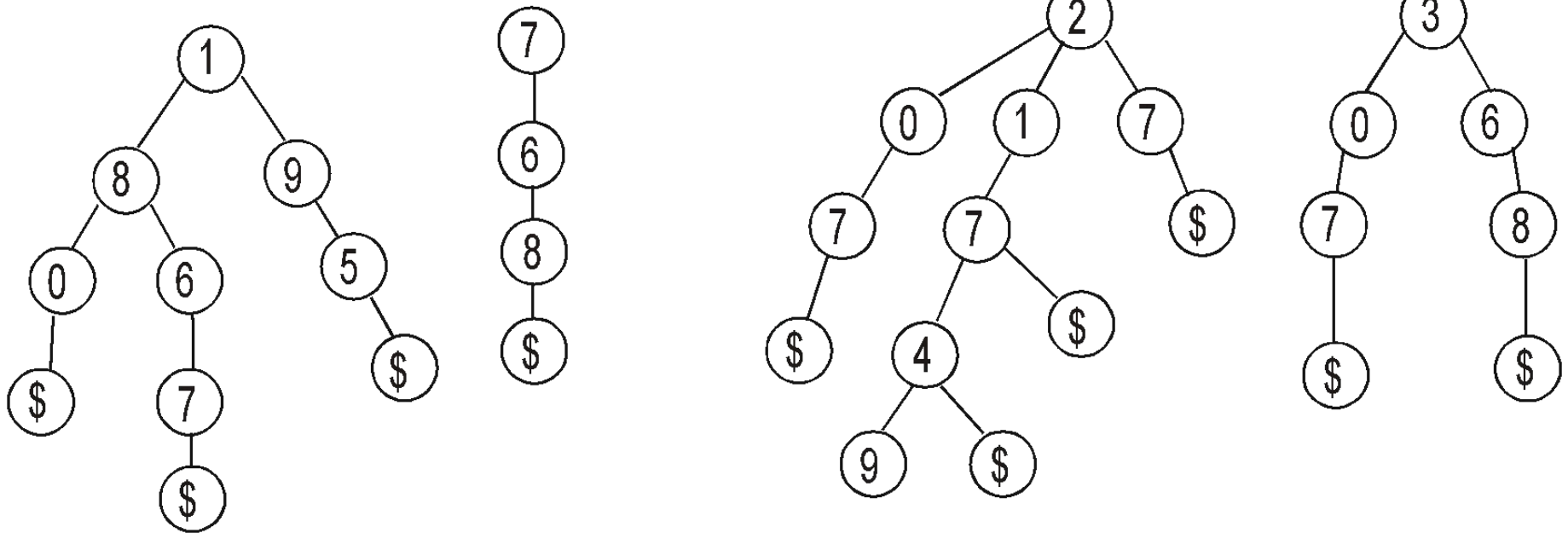
- Очевидное преимущество n-арных деревьев - их меньшая высота, чем у бинарных деревьев, а соответственно и меньшее время поиска
- очевидный недостаток – необходимость в дополнительной памяти для указателей.

Деревья цифрового поиска

- *Деревья цифрового поиска (digital search trees - DST) представляют собой n-арные деревья, ветвление в которых выполняется не в соответствии с результатом сравнения полных ключей, а в соответствии с выбранными разрядами ключа.*

Пример

180, 195, 1867, 768, 207, 217, 2174, 21749, 27, 307, 368;



Если ключи состоят из символов алфавита, то каждая буква определяет некоторую ветвь дерева. Специальный символ в конце каждого листа должен содержать указатель на запись

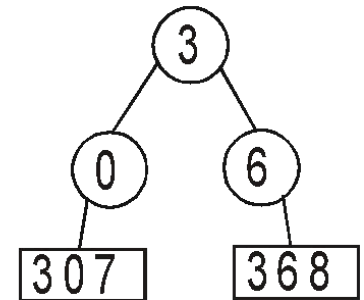
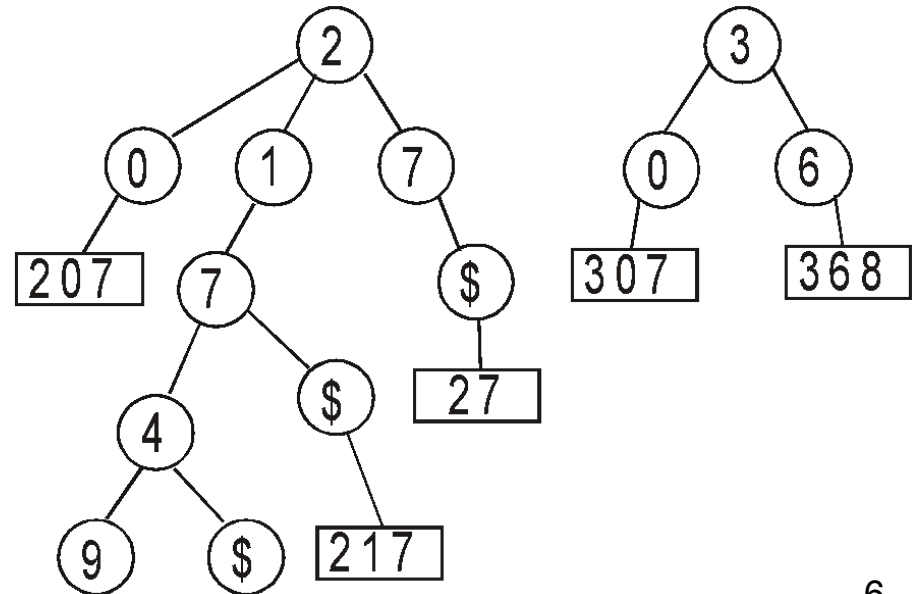
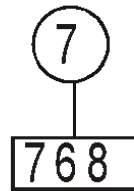
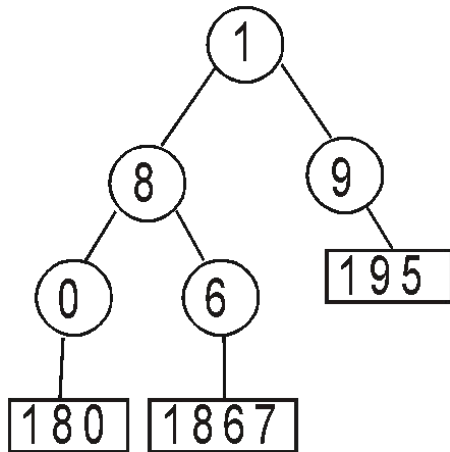
Анализ деревьев цифрового поиска

- Производительность для худшего случая деревьев, построенных по методу поразрядного поиска, значительно выше производительности для худшего случая бинарных деревьев, при условии - если количество ключей велико, а длина ключей мала по сравнению с их количеством.
- Количество сравнений никогда не превышает количество разрядов в ключе поиска.
- Большие DST-деревья, образованные случайными ключами, часто почти идеально сбалансированы.
- Алгоритм поиска и вставки с анализом приведен у Кнута
- Можно организовать поиск только по небольшому списку сыновей для того, чтобы узнать, появляется ли заданный символ в некоторой заданной позиции ключей.

Уплотненный лес

- Данные деревья можно сделать еще меньше, исключив те узлы, из которых можно достичь только одиночные листья. Такое представление называется *уплотненным лесом*.

- 180, 195, 1867, 768, 207, 217, 2174, 21749, 27, 307, 368;



Модификация цифрового дерева

- Если набор ключей является плотным внутри множества всех возможных ключей, то процесс поиска по цифровому дереву неэффективен. Так, если в отделе кадров университета в качестве ключа использовать номер зачетной книжки, то затраты на поиск будут очень большими.
- В этом случае проводится модификация цифрового дерева. Таблица ключей представляется не как дерево, а как некоторый двумерный массив (“БОР”).
- Каждая строка этого массива представляет один из возможных символов, который может появляться в ключе, а каждый столбец представляет узел в цифровом дереве.
- Каждый элемент такого массива является указателем или на другой столбец в этом массиве, или на некоторый ключ и его запись.

Представление уплотненного леса с помощью двумерного массива (“БОРа”)

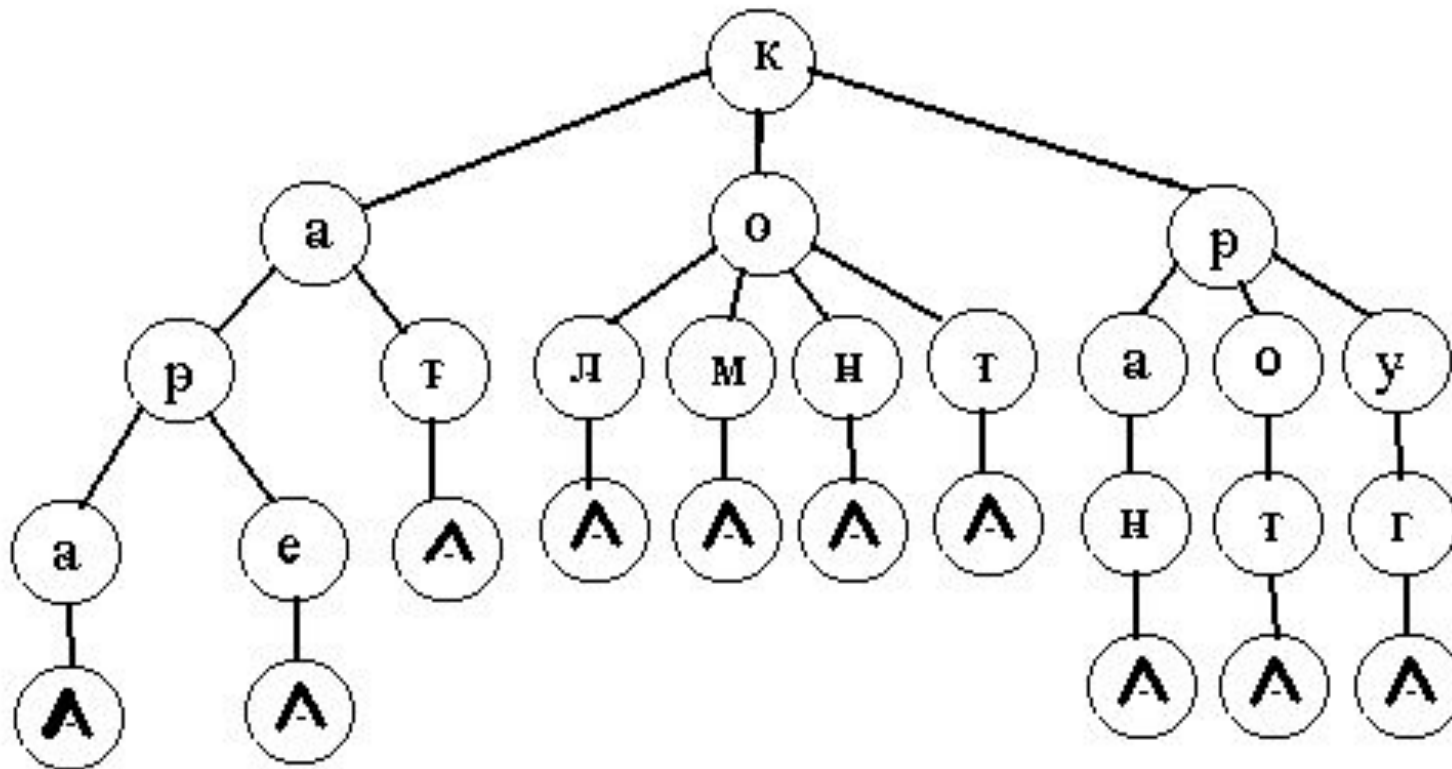
180, 195, 1867, 768, 207, 217, 2174, 21749, 27, 307, 368;

	1	2	3	4	5	6	7	8	9	10	11
0			180	207				307			
1	(2)			(5)							
2	(4)										
3	(8)										
4						(7)					
5											
6			1867					368			
7	768				(6)						
8		(3)									
9		195					21749				
\$				27		217	2174				8

Тrie-деревья (нагруженные деревья)

- Обычно в узлах дерева поиска хранятся значения ключей, но в случае, когда ключами являются достаточно короткие слова, можно рассматривать каждый ключ как список букв, а все списки вместе—как дерево поиска, структура которого несколько отличается от рассмотренных ранее. В этой структуре узлу $(i+1)$ -го уровня ставится в соответствие i -я буква слова, так что каждый узел содержит только один символ. Методы поиска по такому дереву часто весьма экономичны как по памяти, так и по времени

Пример Trie-дерева



- trie-деревья представляют собой структуры данных, применение которых не уступает по эффективности методам хеширования.

Достоинства нагруженных деревьев

- К достоинствам нагруженных деревьев можно отнести возможность перемещения по дереву и выполнения различных операторов за время, пропорциональное длине «обслуживаемого» слова.
- Альтернатива – хеширование. Хеш-функция, чтобы быть действительно «случайной», хеширует каждый символ слова. И, конечно, время вычисления хеш-функции не включает время, необходимое для разрешения коллизий или выполнения операций вставки, удаления или поиска. Поэтому мы вправе ожидать, что нагруженные деревья будут работать значительно быстрее со словарями, состоящими из символьных строк, чем хеш-таблицы.
- Другим достоинством нагруженных деревьев является то, что, в отличие от хеш-таблиц, они поддерживают эффективное выполнение оператора MIN

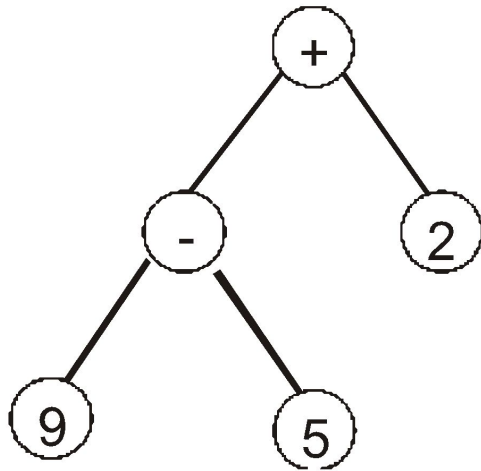
Patricia – деревья

- Основанный на trie-деревьях поиск обладает двумя недостатками:
 - однонаправленное ветвление приводит к созданию дополнительных узлов в trie-дереве, что кажется необязательным;
 - в trie-дереве присутствуют два различных типа узлов, что приводит к усложнениям.
- В 1968 г. Моррисон (Morrison) изобрел способ ликвидации обеих проблем путем применения метода, который назвал *patricia* (practical algorithm to retrieve information coded in alphanumeric - практический алгоритм получения информации, закодированной алфавитно - цифровыми символами).
- Точный анализ среднего случая patricia-дерева сложен; из него следует, что в среднем в patricia-дереве требуется на одно сравнение меньше, чем в стандартном trie-дереве

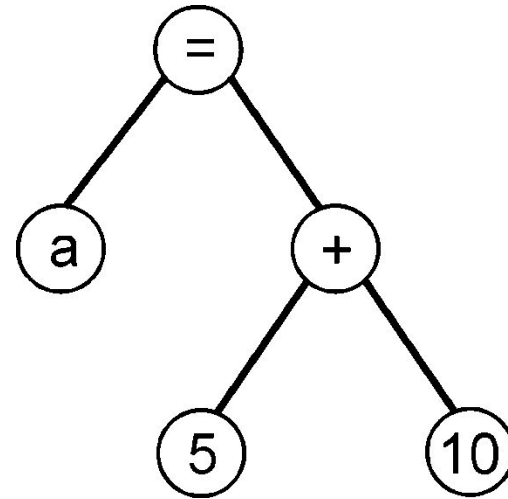
Синтаксические деревья

- Схема грамматического разбора отображаемая синтаксис предложения в форме дерева называется *синтаксическим деревом*.
- С другой стороны, дерево, в котором каждый узел представляет оператор, а потомки этого узла представляют операнды, называется (абстрактным) *синтаксическим деревом*.

Пример синтаксического дерева для арифметического выражения



$9 - 5 + 2$



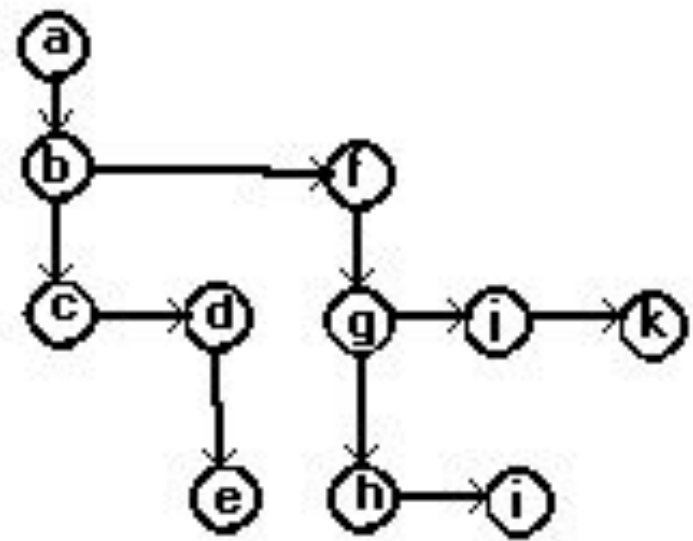
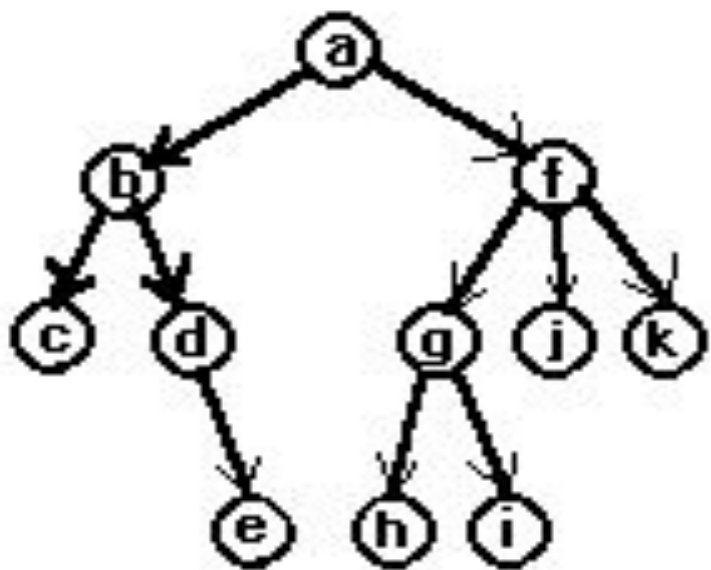
$a = 5 + 10$

Более подробно эти деревья будут рассмотрены в дисциплине «Теория языков программирования и методы трансляции»

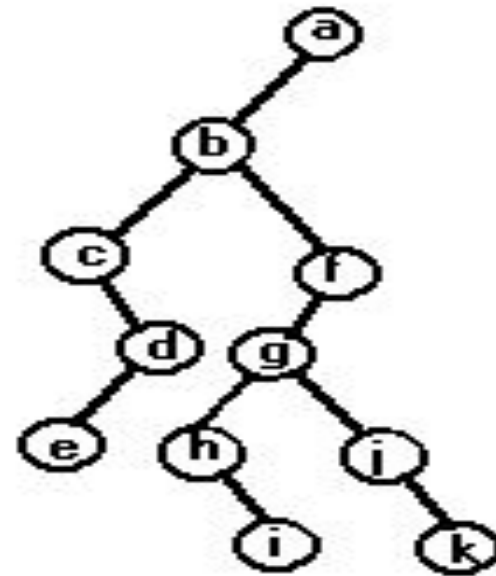
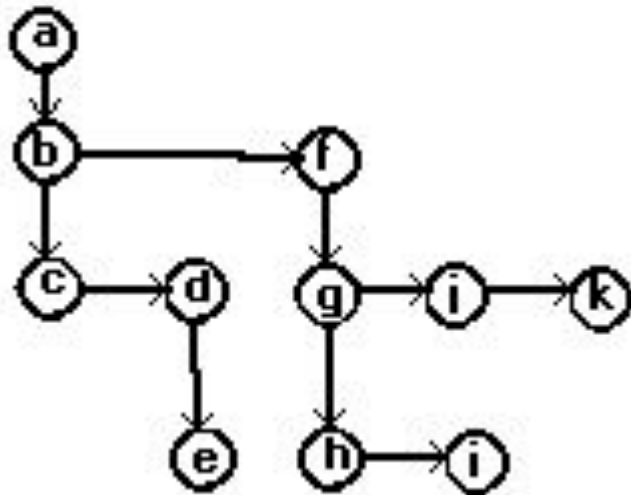
Преобразование упорядоченного n -арного дерева в двоичное дерево

- Любое дерево может быть единственным образом представлено бинарным деревом. Более того, целый лес может быть представлен в виде некоторого бинарного дерева
- Преобразование осуществляется в два этапа

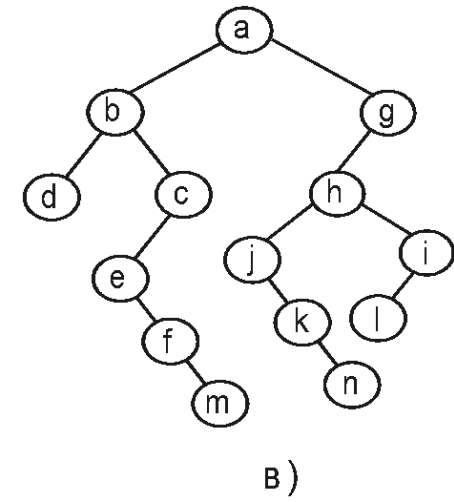
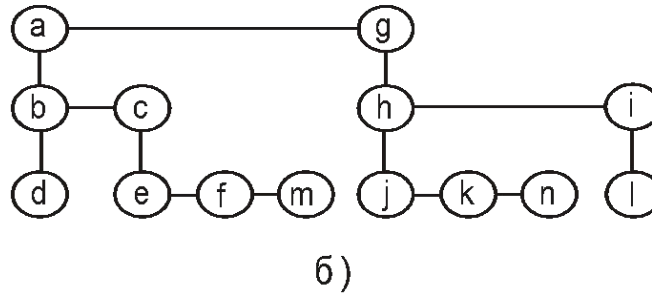
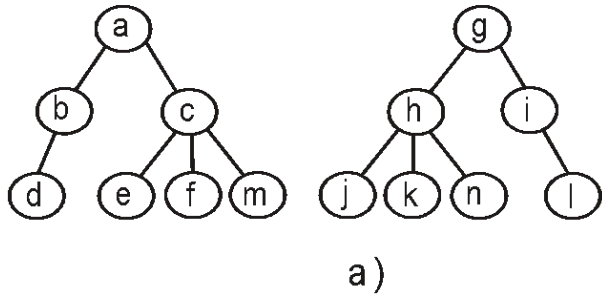
На первом этапе для каждого узла уничтожаются все исходящие из нее ребра, кроме самого левого ребра. Все узлы, имеющие правых братьев, соединяются.



На втором этапе для каждой вершины дерева осуществляется выбор ее левого и правого сыновей по следующему правилу. Левым сыном является вершина, расположенная непосредственно ниже данной вершины, а правым сыном - вершина, расположенная непосредственно справа от данной, и на одном ярусе с ней.



Представление леса в виде бинарного дерева



- а) - заданный лес;
- б) - первый этап преобразования;
- в) - второй этап