



Основы C#

Лекция 7

Бинарные деревья

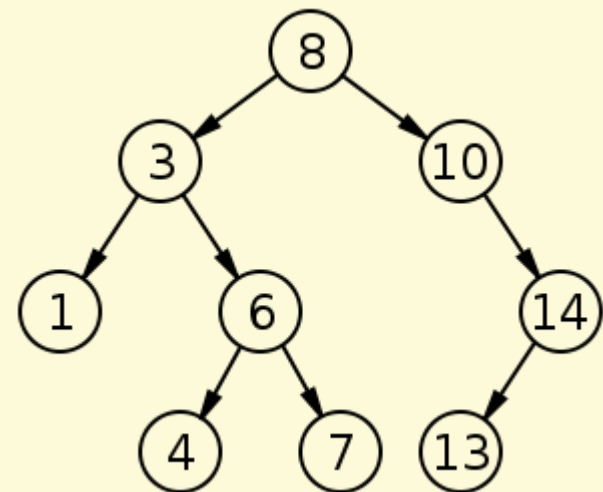


БИНАРНОЕ ДЕРЕВО

Понятие бинарного дерева

Эта динамическая структура данных состоит из узлов (вершин), каждый из которых содержит, кроме данных, не более двух ссылок на различные бинарные деревья (поддеревья, ветви). На каждый узел (вершину) имеется ровно одна ссылка. Начальный узел (вершина) называется корнем дерева.

Дерево является рекурсивной структурой данных, поскольку каждое поддерево также является деревом



Терминология

Узел(вершина) дерева – каждый элемент дерева.

Корень дерева- начальный узел.

Ветви дерева – вершины дерева соединенные направленными дугами

Лист – вершина, не имеющий поддеревьев.

Предок – исходящий узел.

Потомок – входящий узел.

Высота дерева – количество уровней(слоев), на которых располагаются вершины.

Поддерево – часть древообразной структуры данных, которая может быть представлена в виде отдельного дерева.

Степень вершины – количество дуг, которое из нее выходит. Степень дерева равна максимальной степени вершины, входящей в дерево

Уровень вершины – это количество дуг от корня дерева до вершины.

Терминология


Бинарное (двоичное) дерево – это дерево, в котором каждая вершина имеет не более двух потомков.

Неполное бинарное дерево – это дерево, уровни которого заполнены не полностью.

Полное бинарное дерево – это дерево, которое содержит только полностью заполненные уровни.

Почти сбалансированное дерево – это дерево, у которого длины всевозможных путей от корня к внешним вершинам отличаются не более, чем на единицу.

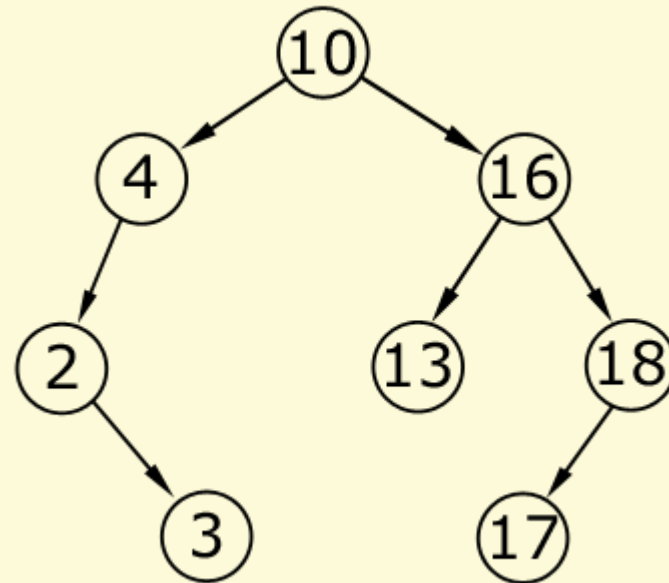
Строгое бинарное дерево – это дерево, у которого вершины имеют степень ноль (у листьев) или два (у узлов).



БИНАРНОЕ УПОРЯДОЧЕННОЕ ДЕРЕВО

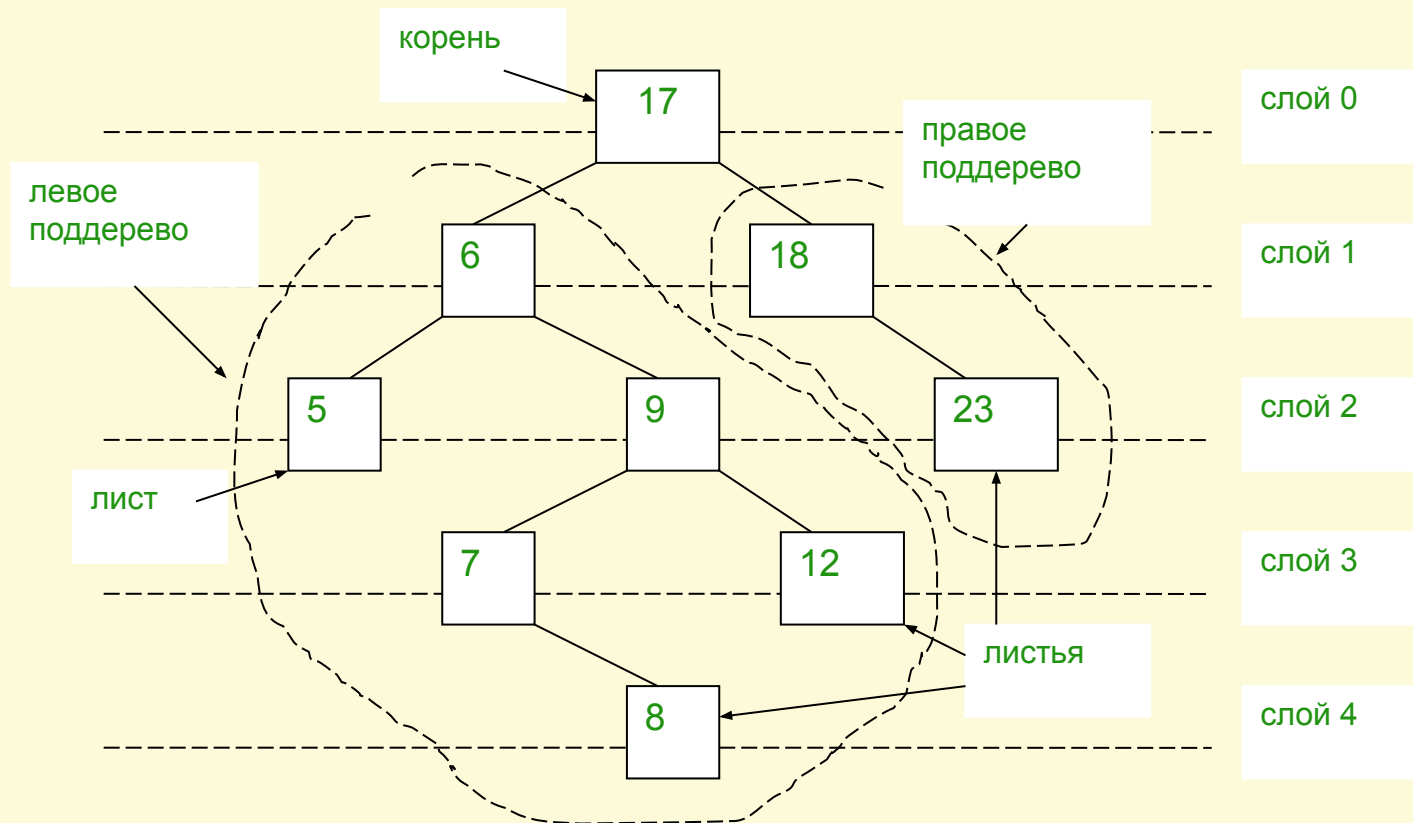
Бинарное упорядоченное дерево (дерево поиска)


Если дерево организовано таким образом, что для каждого узла все ключи его правого поддеревя больше ключа этого узла, а все ключи его левого поддеревя – меньше, то такое дерево называется упорядоченным, или деревом поиска. Одинаковые ключи не допускаются.



Пример построение дерева поиска

Бинарное упорядоченное дерево по приходящим числам: 17, 18, 6, 5, 9, 23, 12, 7, 8.





ИДЕАЛЬНО СБАЛАНСИРОВАННОЕ ДЕРЕВО

Идеально сбалансированное дерево

Идеально сбалансированным деревом - это дерево, в котором число узлов (вершин) в левом и правом поддеревьях отличается не более чем на единицу.

Идеально сбалансированное дерево

Алгоритм равномерного распределения для известного числа вершин формулируют, используя рекурсию.

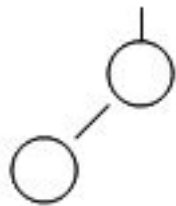
1. Взять одну вершину в качестве корня.
2. Построить тем же способом левое поддереву с $nl=n/2$ вершинами.
3. Построить тем же способом правое поддереву с $nr=n-nl-1$ вершинами.

Идеально сбалансированное дерево

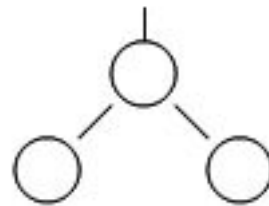
$n = 1$



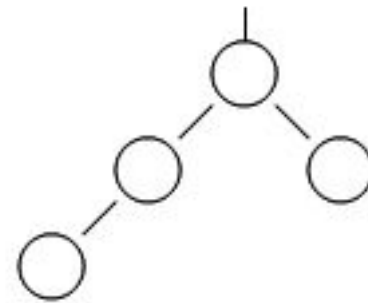
$n = 2$



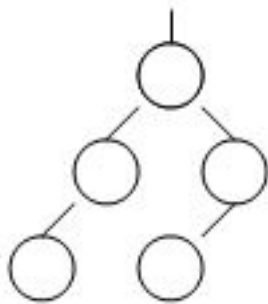
$n = 3$



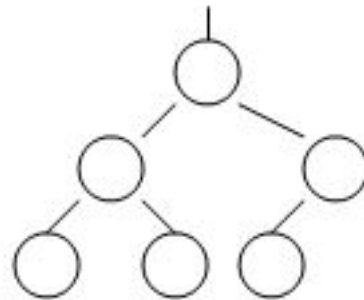
$n = 4$



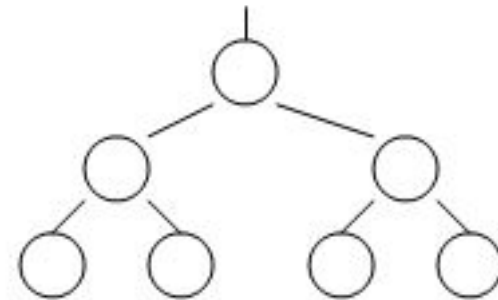
$n = 5$




$n = 6$



$n = 7$



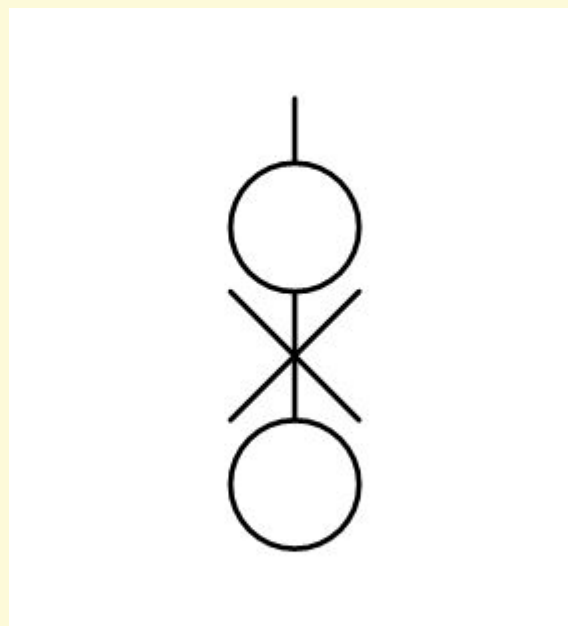


ОПЕРАЦИИ С БИНАРНЫМ УПОРЯДОЧЕННЫМ ДЕРЕВОМ

1.УДАЛЕНИЕ УЗЛА ИЗ ДЕРЕВА

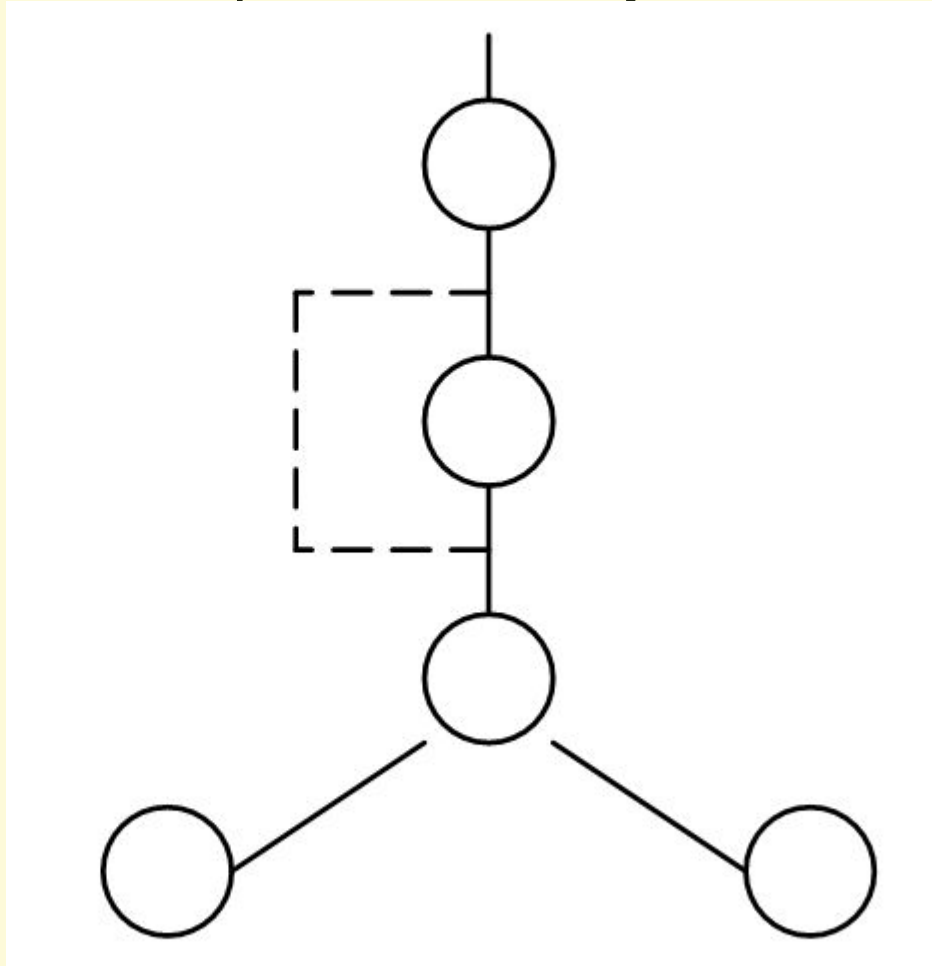
Исключаемый узел – лист.

Действия: Удалить ссылку на данный узел.



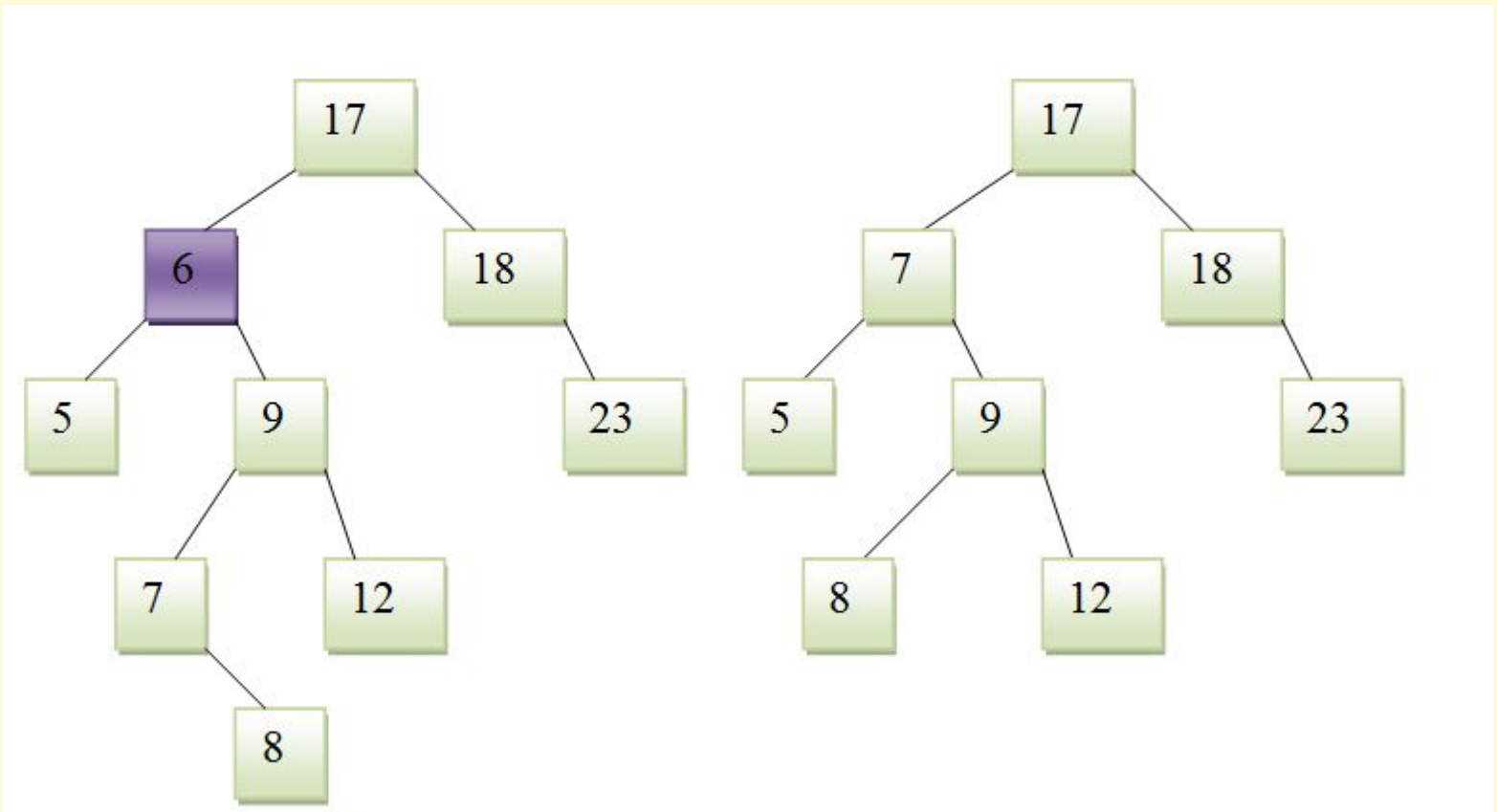
Из исключаемого узла выходит одна ветвь.


Действия: переназначить указатель (пунктир).



Из исключаемого узла выходят две ветви

Действия: на место удаляемого узла надо поставить либо самый правый узел левой ветви, либо самый левый узел правой ветви для сохранения упорядоченности дерева.





ОПЕРАЦИИ С БИНАРНЫМ УПОРЯДОЧЕННЫМ ДЕРЕВОВОМ

2. ОБХОД (ПРОСМОТР) ДЕРЕВА

Обходы (просмотры) дерева

Обход дерева – это упорядоченная последовательность вершин дерева, в которой каждая вершина встречается только один раз.

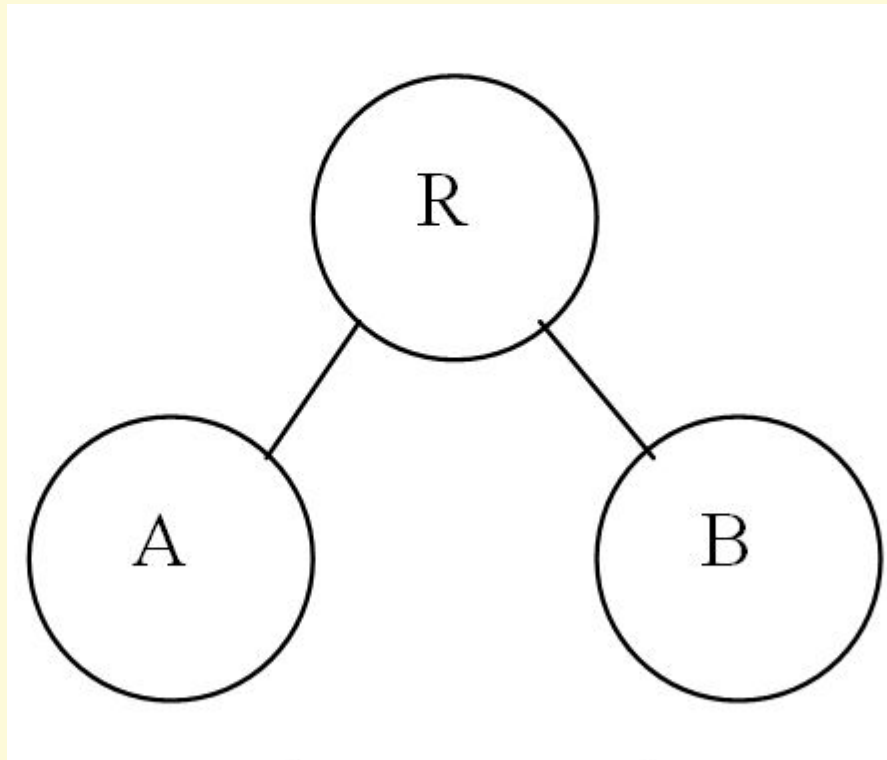
Способы обхода (просмотра) дерева

а) Просмотр слева-направо: ARB (левое поддерево-корень-правое поддерево).

б) Просмотр сверху-вниз: RAB (корень до поддерева).

в) Просмотр снизу-вверх: ABR (корень после поддерева).

г) Просмотр справа-налево: BRA (правое поддерево-корень-левое поддерево).

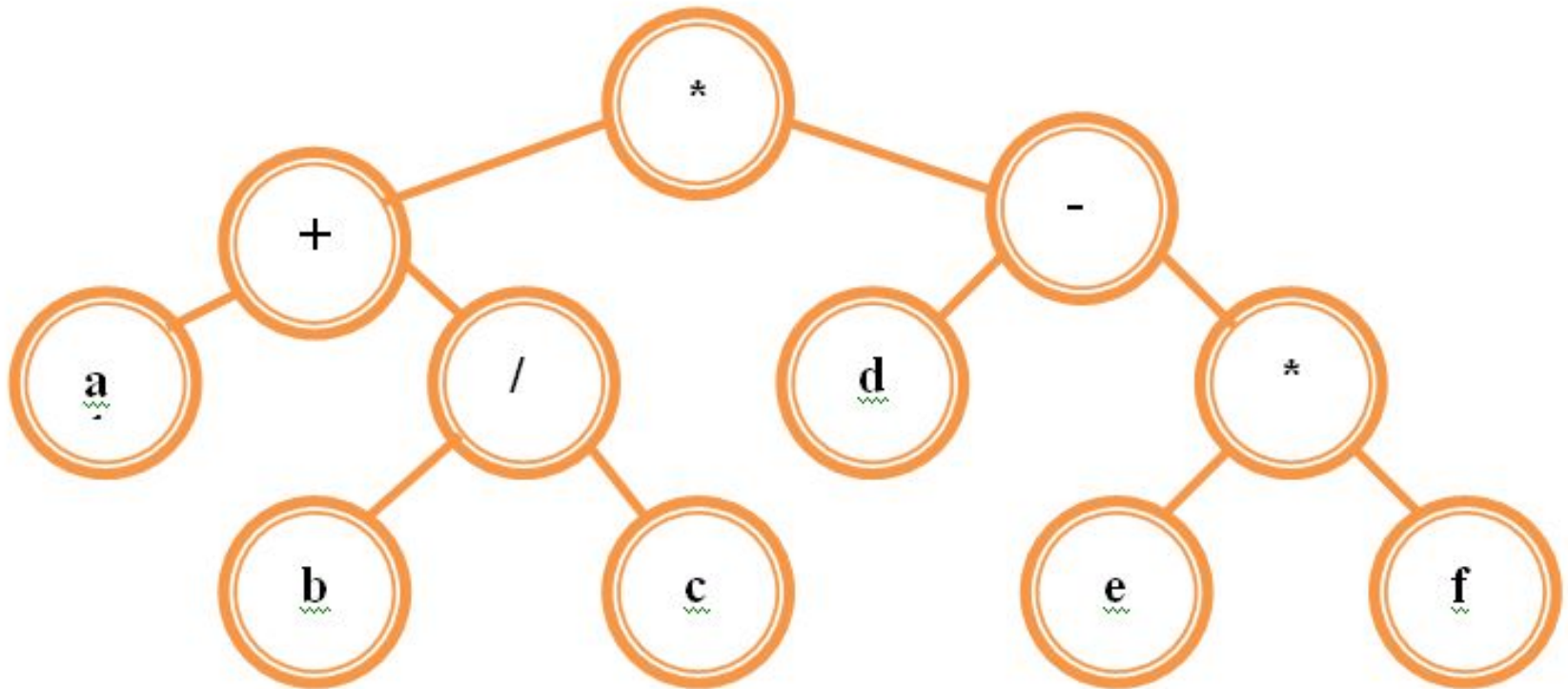


Пример обхода дерева

а) ARB: - инфиксная запись. $(a + b / c) * (d - e * f)$

б) RAB: - префиксная запись. $* + a / b c - d * e f$

в) ABR: - постфиксная запись. $abc / + def * - *$



Пример обхода дерева

а) ARB: - 5, 6, 8, 9, 11, 12, 17, 18, 23.

б) BRA - 23, 18, 17, 12, 11, 9, 8, 7, 6, 5.

