Алгоритмы и структуры данных поиска

ДЗ №4, задача 4-a. Теоретическая часть.

*Описание алгоритма.*

На вход подается

n – число элементов в дереве.

n элементов дерева В порядке preorder.

Необходимо вывести ключи в порядке inorder, postorder.

Алгоритм строится следующим образом. Сначала строим из этих элементов BST за линейное время. Затем известными процедурами выводим ключи в порядке inorder, postorder.

Рассмотрим линейный алгоритм построения BST из ключей, распложенных в порядке preorder:

Заметим, что первый элемент всегда является корнем, поскольку обход начинается всегда с него. Далее содержатся подряд элементы принадлежащие левому поддереву, они все меньше или равны корня (первого элемента). А сразу после них содержатся подряд элементы, которые содержит правое поддерево, они все больше корня. Причем если рассмотреть отдельно эти наборы левого и правого поддерева, то они тоже будут описывать соответствующее поддерево в порядке preorder. Исходя из этих рассуждений можно предложить следующий алгоритм. Начинать будем с корня, то есть с первого элемента, и установим для него отрезок , в который он всегда будет входить. Для левого поддерева установим отрезок , и все ключи попадающие в него будут являться частью правого поддерева. Аналогично для правого поддерева .

Такой алгоритм будет работать рекурсивно за линейное время.

Далее, построенное дерево, обойдем известными процедурами inorder\_tree\_walk(root), postorder\_tree\_walk(root), каждая из которых работает за линейное время.

*Сложность.*

Построение дерево займет времени, как и каждый из обходов.

Cложность алгоритма O(n).

*Память.*

Потребуется считать элементы в массив, которой займет памяти. Затем, построить с помощью него дерево, которое будет занимать в памяти также .

Итого расходы по памяти будут занимать

Махотин Иван, ШАД.