Федер Евгений, Домашнее задание №1

Задание 1.

Для начала научимся ставить ключ в корень дерева поиска фиксированной структуры из какого-то отсортированного множества размера n. Пусть у корня l вершин слева и r вершин справа($l+r+1 = n$). Тогда поставим в корень $l+1$-ый элемент множества.

Чтобы построить дерево поиска будем делать это рекурсивно, постепенно запускаясь от левого и правого потомка текущего корня(множества будет делиться: для левого - первые l элементов, для правого - последние r элементов). Такое построение выполняется и будет верным так как:

1) Выполняется главное свойство дерева поиска. Вершины слева меньше текущего ключа и веришны справа больше(это следует из отсортированности множества).

2) по построению структура будет такой же и построение не зациклится так как на каждом шаге мы уменьшаем количество элементов множества на 1.

Так как на каждом шаге не возникает никаких двоякостей дерево строится единственным образом. *Что и требовалось доказать*

Задание 2.

Рассмотрим задачу с конца. Пусть у нас есть n свободных вершин(листов) и мы можем соединять их между собою создавая новую вершину. При этом если у тебя было n листов, то их станет $n - k + 1$, где k - количество соединенных вершин. В конце должен остаться один лист.

Таким образом мы сделаем как максимум $m - 1$ операций соединения. А максимальное количество вершин при этом будет если мы будем соединять вершины только по 2(то есть $2 * m - 1$ вершин), а это следует то, что мы хотели.

Задание 3.

Построим дерево, чтобы оно не противоречило условию по индукции:

База: Для одной вершины $h = 0$ и утверждение выполняется

Инд.переход: так как дерево двоичное, то у нас может быть два случая:

1) У нас есть одно дерево, для которого выполнено утверждение. Мы подвешиваем его к новой вершине. Следовательно глубина всех листьев увеличилась на 1 и сумма стала в 2 раза меньше.

2) У нас есть два дерева, для которых выполнено утверждение. Мы подвешиваем их к новой вершине. Следовательно $h_{new} = h_1/2 + h_2/2 = (h_1 + h_2)/2 \leq 1$

Так как с помощью таких преобразований мы можем построить любое дерево из m листьев, то утверждение доказано. Как видно из индукции равенство будет дотигаться, когда у каждой вершины дерева будет по два ребенка(кроме листьев).

Задание 5.

Пусть дано два дерева T_1 и T_2, где $\forall a \in T_1 \, \forall b \in T_2 \, a.key \leq b.key$. Также не умоляя общности пусть $h(T_1) \geq h(T_2)$ и $b_{min} = m$.

Пойдем по дереву T_1 вправо, пока не наткнемся на элемент $x : h(x) = h(T_2)$. Теперь сделаем такое преобразование:

$p(x).r = m$

$m.l = x$

$m.r = T_2.root$

Теперь смотрим, что получилось с высотами: При удалении вершины из меньшего дерева его высота могла уменьшится. Ну сделаем тогда балансировку в вершине x за $O(1)$

По времени получаем $O(log(n)) + O(1) = O(log(n))$