HW1

Shane Drafahl

30 August,2017

1. Use mathematical induction to prove the following statements

(a)
$\forall_n \geq 1, \sum_{i=1}^{n} i^3 = (\sum_{i=1}^{n} i)^2$

lemma: $\sum_{i=1}^{n} i = \frac{n^2}{2} + \frac{n}{2}$ where $n \in N$
Base Case: $\sum_{i=1}^{1} i = 1 = \frac{1^2}{2} + \frac{1}{2}$
Inductive Hypothesis: $\sum_{i=1}^{n} i = \frac{n^2}{2} + \frac{n}{2}$
Inductive Steps:
Using induction on n
$\frac{(n+1)^2}{2} + \frac{n+1}{2} = \frac{n^2}{2} + \frac{3n}{2} + 1 = \frac{n^2}{2} + \frac{n}{2} + (n+1)$

So therefore the lemma is true for all natural numbers n greater than 1.

With the lemma we can solve the proof easier

Base Case: For n = 1 would be $\sum_{i=1}^{1} i^3 = 1^3 = 1$. $(\sum_{i=1}^{1} i)^2 = 1$.

Inductive Hypothesis: From the base n is true for $\forall_n \geq 1, \sum_{i=1}^{n} i^3 = (\sum_{i=1}^{n} i)^2$

Inductive Step: Using lemma we can re-write this to $(\frac{n^2}{2} + \frac{n}{2})^2 = \sum_{i=1}^{n} i^3$
We will first add n+1, $(\frac{(n+1)^2}{2} + \frac{(n+1)}{2})^2 = \frac{n^4}{4} + \frac{3n^3}{2} + \frac{13n^2}{4} +$ 3n + 1 . Now we will do $(\frac{n^2}{2} + \frac{n}{2})^2 + (n+1)^3 = \frac{n^4}{4} + \frac{3n^3}{2} + \frac{13n^2}{4} +$ 3n + 1 . So therfore $\forall_n \geq 1, \sum_{i=1}^{n} i^3 = (\sum_{i=1}^{n} i)^2$ is true for all natural numbers for n greater than or equal to 1.

(b) $\forall_n \geq 4, 2^n < n!$

Basis: for n = 4 2^4 = 16 and 4! = 24

Inductive Hypothesis: $\forall_n \geq 4$, $2^n < n!$ is true for n.

Inductive Step: $2^{n+1} = 2^n * 2$. (n + 1)! = n! * (n + 1) Since we know from the inductive hypothesis that $\forall_n \geq 4$, $2^n < n!$ so for this to remain true the right side and the left would need to be multiplied by the same value or the right would need to be multiplied by a greater value. The left is multiplied by 1 and the right is multiplied by (n + 1). We know that n has to be 4 or greater so we know (n + 1) ¿ 2 so therfore $\forall_n \geq 4$, $2^n < n!$ for all natural numbers n.

2. Refer to the definition of Full Binary Tree from the notes. For a Full Binary Tree T, we use n(T), h(T), i(T) and ‘(T) to refer to number of nodes, height, number of internal nodes (non-leaf nodes) and number of leaves respectively. Note that the height of a tree with single node is 1 (not zero). Using structural induction, prove the following:

(a) For every Full Binary Tree T, n(T) ≥ h(T).

Basis: For a tree T with a single root node has a height 1. It has a single node and 1 = 1 so for n(T) ≥ h(T) is true.

Inductive Hypothesis: n(T) ≥ h(T) is true is true for single root node tree T.

Inductive Step: Suppose you have a tree T that n(T) ≥ h(T) is true. If we want to add more nodes to the tree and for it to also be a full binary tree we must add two nodes per leaf node. The root node doesnt have a parent node so its child nodes are the first nodes to branch into two different child nodes. So starting from the roots direct child node one layer difference thus $(h(T) - 1)^2$ equals the number of leaf nodes. So increase the height of the tree by 1 would require two nodes for every leaf node thus for the tree with increased height T_a. So h(T_a) = h(T) + 1 and n(T_a) = n(T) * 2. So n(T) * 2 ≥ h(T) + 1 is true by the inductive hypothesis because we already know n(T) ≥ h(T) and n(T) is being multiplied by 2 and h(T) is only added by 1. So the left side which is bigger is being increased more than the right so therfore n(T) ≥ h(T) is true for all complete binary search trees T.