Homework 1

Yu-Ju Teng   Ling-Hsuan Chen

Question1

1. The Harmonic series $H(k)$ is defined by $H(k) = 1 + \frac{1}{2} + \frac{1}{3} + \cdots + \frac{1}{k-1} + \frac{1}{k}$. Prove that $H(2^n) \geq 1 + \frac{n}{2}$, for all $n \geq 0$ (which implies that $H(k)$ diverges).

Question1

[Claim] For every $n \geq 0$, $\mathrm{H}(2^n) \geq 1 + \frac{n}{2}$

[Base Case] (n=0) $H(2^0) = H(1) = 1 \ \geq \ 1 + \frac{0}{2}$

[Inductive Step] $(n > 0)$

$$H(2^n) = \underline{1 + \frac{1}{2} + \frac{1}{3} + \ldots + \frac{1}{2^{n-1}}} + \frac{1}{2^{n-1}+1} + \ldots + \frac{1}{2^n}$$

[By Inductive Hypothesis]

$$\geq 1 + \frac{n-1}{2} + \left(\frac{1}{2^{n-1}+1} + \frac{1}{2^{n-1}+2} + \ldots + \frac{1}{2^n}\right)$$

$$\geq 1 + \frac{n-1}{2} + \left(\frac{1}{2^n} + \frac{1}{2^n} + \ldots + \frac{1}{2^n}\right)$$

$$= 1 + \frac{n-1}{2} + \left(2^{n-1} \times \frac{1}{2^n}\right)$$

$$= 1 + \frac{n-1}{2} + \frac{1}{2}$$

$$= 1 + \frac{n}{2}$$

Question2

2. Consider *proper* binary trees, where every internal (non-leaf) node has two children. For any such tree T, let l_T denote the number of its leaves and m_T the number of its internal nodes. Prove *by induction* that $l_T = m_T + 1$.

Question2

[Claim] For any proper binary tree T, let l_T denote the number of its leaves and m_T the number of its internal nodes, $l_T = m_T + 1$

[Base Case] ($m_T = 0$) There's only one node (root) in the tree, and the node is a leaf. $1 = 0 + 1$.

[Inductive Step] ($m_T > 0$)

[Inductive Hypothesis] ($0 < m_T < n$) For every proper binary tree with less than n internal nodes, the equation holds.

Assume there's a proper binary tree T with l_T and $m_T = n$. We pick 1 internal node with 2 leaves and delete its leaves and the internal node turns to a leaf. Now we have a tree T' with $l'_T = l_T - 1$ and $m'_T = m_T - 1 < n$. By Inductive Hypothesis, $l'_T = m'_T + 1$.

$l_T = l'_T + 1 = m'_T + 1 + 1 = m_T + 1$

Question3

3. (2.7) Given a set of $n+1$ numbers out of the first $2n$ (starting from 1) natural numbers 1, 2, 3, . . ., $2n$, prove that there are two numbers in the set, one of which divides the other.

Question3

[Base Case] $(n = 1)$

When $n = 1$, the selection set is $\{1, 2\}$, and 2 can be divided by 1.

[Inductive Step] $(n = k > 1)$

Given a set of $k + 1$ numbers out of the first $2k$ numbers, the first $2k$ numbers are $\{1, 2, ..., 2k - 3, 2k - 2, 2k - 1, 2k\}$.
[Inductive Hypothesis] $(n = k - 1)$ Given a set of k numbers out of the first $2k - 2$ numbers, there are two numbers in the set, one of which divides the other.
The first $2k - 2$ numbers are $\{1, 2, ..., 2k - 3, 2k - 2\}$.

1. If both $2k - 1$ and $2k$ are not in the selection set, there are $k + 1$ numbers being selected in the first $2k - 2$ numbers. By Inductive Hypothesis we proved it.
2. If one of $2k - 1$ and $2k$ is in the selection set, there are k numbers being selected in the first $2k - 2$ numbers. By Inductive Hypothesis we proved it.

Question3 (Continue)

3. If both of $2k-1$ and $2k$ are in the selection set, there are $k-1$ numbers being selected in the first $2k-2$ numbers, consider following two cases :
   1. If k is in the selection set, then k divides $2k$.
   2. If k is not in the selection set:
      1. If there are two numbers in the set out of the first $2k-2$ numbers (excluding k) and one of which divides the other, then there will be no problem.
      2. If not, when we put k back to the set out of first $2k-2$ numbers, by Inductive Hypothesis, there should be two numbers in the set, one of which divides the other. Thus, there is at least 1 number can divide k in the set, and this number can divide $2k$ as well.

Now we prove all cases. By mathematical induction, the claim is true.

Question4

4. (2.37) Consider the recurrence relation for Fibonacci numbers $F(n) = F(n-1)+F(n-2)$. Without solving this recurrence, compare $F(n)$ to $G(n)$ defined by the recurrence $G(n) = G(n-1)+G(n-2)+1$. It seems obvious that $G(n) > F(n)$ (because of the extra 1). Yet the following is a seemingly valid proof (by induction) that $G(n) = F(n)-1$. We assume, by induction, that $G(k) = F(k)-1$ for all k such that $1 \leq k \leq n$, and we consider $G(n+1)$:

$$G(n+1) = G(n) + G(n-1) + 1 = F(n) - 1 + F(n-1) - 1 + 1 = F(n+1) - 1$$

What is wrong with this proof?

Question4

The correctness of base case wasn't proved.
That is, the inductive hypothesis may not hold at the very beginning.
Moreover, there is no clear definition for G(n).

$$\mathrm{F(n)} = \begin{cases} 1 & \text{if } n = 1. \ [\textit{Base case}] \\ 1 & \text{if } n = 2. \ [\textit{Base case}] \\ F(n-1) + F(n-2) & \text{if } n \geq 3. \ [\textit{Inductive step}] \end{cases}$$

$$\mathrm{G(n)} = \begin{cases} g_1 & \text{if } n = 1. \ [\textit{Base case}] \\ g_2 & \text{if } n = 2. \ [\textit{Base case}] \\ G(n-1) + G(n-2) + 1 & \text{if } n \geq 3. \ [\textit{Inductive step}] \end{cases}$$

As a result, we cannot prove the inductive hypothesis.

Question4

If we set $G(1) = g_1 = 0$ and $G(2) = g_2 = 0$, then the base cases $G(1) = F(1) - 1$ and $G(2) = F(2) - 1$.
In this case, the proof will be correct.

If we set $G(1) = g_1 = 1$ and $G(2) = g_2 = 1$, then the base cases $G(1) \neq F(1) - 1$ and $G(2) \neq F(2) - 1$.
In this case, the proof will be wrong.

Question5

(a) (10 points) Define inductively a function Max that determines the largest of all key values of a binary tree. Let $Max(\perp) = 0$, though the empty tree does not store any key value. (Note: use the usual mathematical notations; do not write a computer program.)

(b) (10 points) Suppose, to differentiate the empty tree from a non-empty tree whose largest key value happens to be 0, we require that $Max(\perp) = -1$. Give another definition for Max that meets this requirement; again, induction should be used somewhere in the definition.

Question5 (a)

Let T be a tree, the function Max is defined as follows:

$$Max(T) = \begin{cases} 0 & \text{if } T = \perp. \text{ [}\textit{Base case}\text{]} \\ \mathbf{max}(k, \mathbf{max}(Max(t_l), Max(t_r))) & \text{if } T = node(k, t_l, t_r), \text{ [Inductive step]} \end{cases}$$

, where $\mathbf{max}(\mathrm{x}, \mathrm{y}) = \begin{cases} x & \text{if } x > y \\ y & \text{otherwise.} \end{cases}$

Do not write pseudo-code unless the question requires.

Question5 (b)

- Every key value in the binary tree with non-negative integer key values are larger than -1.
- Hence, the result of max function won't be affected by any empty node.
- We can apply the same inductive step as the former problem.

Let T be a tree, the function Max is defined as follows:

$$Max(T) = \begin{cases} -1 & \text{if } T = \bot.\ [\textit{Base case}] \\ \mathbf{max}(k, \mathbf{max}(Max(t_l), Max(t_r))) & \text{if } T = node(k, t_l, t_r),\ [\text{Inductive step}] \end{cases}$$

, where $\mathbf{max}(\text{x, y}) = \begin{cases} x & \text{if } x > y \\ y & \text{otherwise.} \end{cases}$