Homework 1

林宏陽、周若涓、曾守瑜

Question1

[Base Case] (n=1) $1^3 = 1^2$

[Induction Step] $1^3 + 2^3 + 3^3 + ... + n^3 + (n+1)^3$

$$= (1 + 2 + ... + n)^2 + (n+1)^3$$
$$= (1 + 2 + ... + n)^2 + (n+1)(n+1)^2$$
$$= (1 + 2 + ... + n)^2 + n(n+1)^2 + (n+1)^2$$
$$= (1 + 2 + ... + n)^2 + n(n+1)(n+1) + (n+1)^2$$
$$= (1 + 2 + ... + n)^2 + 2(1 + 2 + ... + n)(n+1) + (n+1)^2$$
$$= (1^2 + 2^2 + ... + n + (n+1))^2$$

Question2

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

[Induction Step] $H(2^{n+1}) = 1 + \frac{1}{2} + \frac{1}{3} + ... + \frac{1}{2^n} + \frac{1}{2^n+1} + ... + \frac{1}{2^{n+1}}$

$$\geq 1 + \frac{n}{2} + (\frac{1}{2^n+1} + \frac{1}{2^n+2} + ... + \frac{1}{2^{n+1}})$$

$$\geq 1 + \frac{n}{2} + (2^n \ * \frac{1}{2^{n+1}})$$

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

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

Question3

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.

[Base Case]($n = 1$) When $n = 1$, the selection set $\{1, 2\}$ is trivial.

[Induction Step](from n to $n+1$)

1. If both $2n+1$ and $2n+2$ are not in the selection set, there are $n+2$ numbers being selected in the first $2n$, simply by I.H.
2. If one of $2n+1$ and $2n+2$ is in the selection set, there are $n+1$ numbers being selected in the first $2n$, simply by I.H.

cont'd

3. If both of $2n+1$ and $2n+2$ is in the selection set, consider following two cases :
   1. If $n+1$ is in the selection set, then $n+1$ divides $2n+2$.
   2. If $n+1$ is not in the selection set, then I.H. tells us one of following is correct :
      1. First $2n$ 選 $n+1$ 個中，能被整除的是 $n+1$, 因此 [First $2n$ 選 $n+1$ 個 $\cup \{2n+1, 2n+2\}$] 中有 $n+1$ 的因數，$n+1$ 的因數能整除 $2n+2$. (不會是 $n+1$ 去整除別人因為沒有任何其他數在 first $2n$ 中能被 $n+1$ 整除)
      2. First $2n$ 選 $n+1$ 個中能互相整除的是 n+1 以外的兩個數，也就是 [First $2n$ 選 $n+1$ 個 $\cup \{2n+1, 2n+2\}$] 中互相整除的兩數。

Now we prove all cases. By induction, I.H. is true.

cont'd

注意

1. 定義每個使用的符號或變數。
2. 如果寫 replace $n+1$ by $2n+2$，解釋為什麼可以這麼做。

Question5

- a **Inductively** define a function *SUM*
  $\rightarrow$ Define a recursive function
  Base case of recursion: $SUM(\perp) = 0$
  if tree is in the form $node(l, t_l, t_r)$,
  $SUM(node(l, t_l, t_r)) = k + SUM(t_l) + SUM(t_r)$

$$SUM(tree) = \begin{cases} 0, & tree = \perp \\ k + SUM(t_l) + SUM(t_r), & tree = node(l, t_l, t_r) \end{cases}$$

Can I Write Pseudocode?

教授說不行，除非題目特別指定要寫 pseudocode
作業一第五題寫 code 或 pseudocode 我不會扣分
作業二第一題就會扣了，因為助教課有提醒過

Question5

- b. Base case: $SUM(\perp) = -1$
  除此之外，其他樹的運算結果需要和 (a) 一樣
  問題是，每棵樹都有很多空的子樹，不能讓這些 -1 影響結果

$$SUM(tree) = \begin{cases} -1, tree = \perp \\ k, tree = node(k, \perp, \perp) \\ k + SUM(t_l), tree = node(k, t_l, \perp) \text{ and } t_l \neq \perp \\ k + SUM(t_r), tree = node(k, \perp, t_r) \text{ and } t_r \neq \perp \\ k + SUM(t_l) + SUM(t_r), otherwise \end{cases}$$

Question5

- (b) 這樣寫，更簡潔！

$$SUM(tree) = \begin{cases} -1, & tree = \bot \\ SUM'(tree), & otherwise \end{cases}$$

$$SUM'(tree) = \begin{cases} 0, & tree = \bot \\ k + SUM'(t_l) + SUM'(t_r), & tree = node(k, t_l, t_r) \end{cases}$$

Question5

- (c) Refine the definition
  1. The empty tree, denoted $\perp$, is a binary search tree.
  2. If t_l and t_r are BST,
     and if t_l is not empty, the key value of t_l is smaller than k,
     and if t_r is not empty, the key value of t_r is larger than k,
     then $node(k, t_l, t_r)$ is also a BST.

雖然題幹就給了一個 Base case，但是你還是需要在答案稍微提及一下
少寫 base case（或完全不提及其存在）2 分
少寫 search 不扣分

Question5

- 常錯的點：沒有提及 empty tree 時要怎麼取 key value，也沒有說遇到 $\perp$ 要忽略，扣 2 分
  常錯的點：寫 $t_l < k < t_r$, type error（沒有定義 tree 與 number 怎麼比大小），扣 4 分
  不要省略"the key value of" $t_l < k <$ "the key value of" t_r