# CSC236H, Winter 2016
# Assignment 2

## Sample Solutions

1. Let $T$ be a set of rooted trees such that for each $t \in T$:

   - nodes are labeled with positive integers. That is, each node $v$ is labeled with an integer $a_v$;
   - if $w$ is a child of $v$, then $a_w < a_v$.

   Prove that for all $t \in T$, the root of $t$ has the largest label of all nodes in the tree.

   **Solution:** We will prove the claim by using induction over the structure of the trees in $T$.
   $P(t)$ : The root node of $t$ has the largest label in $t$.

   **Base Case:** Let $t$ be a single node.
   Then $t$ is the root node and has the largest label. So, $P(t)$.

   **Induction Step:** Let $t \in T$. Let $r$ be the root of a tree $t$.
   Assume that for each sub-tree $t'$ of $t$, $P(t')$, i.e., the root node of $t'$ has the largest label in $t'$. **[IH]**
   **WTP:** $P(t)$, i.e., the root of $t$ has the largest label in $t$.

   Let $r$ be the root node of $t$.
   If $r$ has no children, then $t = r$ and we're done by the base case. Otherwise, let $r_1, ..., r_n$ be *all* the children of $r$, and $t_1, ..., t_n$ be the sub-trees of $t$ with roots $r_1, ..., r_n$.
   By IH, $a_{r_i}$ is the largest label in $t_i$, for all $1 \le i \le n$.
   By the definition of $T$, $a_{r_i} < a_r$, for all $1 \le i \le n$.
   Therefore, $a_r$ is the largest label in $t$, and so $P(t)$ holds.

2. Let $M \subseteq \mathbb{Z}^2$ be a set defined as follows:

   - $(3, 2) \in M$,
   - for all $(x, y) \in M$, $(3x - 2y, x) \in M$,
   - nothing else belongs to $M$.

   Use structural induction to prove that for all $(x, y) \in M$, there exists $k \in \mathbb{N}$, such that

   $$(x, y) = (2^{k+1} + 1, 2^k + 1).$$

   **Solution:** $P((x, y))$ : there exists $k \in \mathbb{N}$ s.t. $(x, y) = (2^{k+1} + 1, 2^k + 1)$.
   The goal is to prove for all $(x, y) \in M$, $P((x, y))$.

**Base Case:** Let $(x, y) = (3, 2)$.
Let $k = 0$.
Then $2^{k+1} + 1 = 2^1 + 1 = 3$ and $2^k + 1 = 2^0 + 1 = 2$.
Then there exists $k \in \mathbb{N}$, such that $(x, y) = (2^{k+1} + 1, 2^k + 1)$ and so $P((x, y))$.

**Induction Step:** Let $(x, y) \in M$. By definition, $M$ includes $(3x - 2y, x)$.
Suppose $P((x, y))$, i.e., there exists $k \in \mathbb{N}$ such that $(x, y) = (2^{k+1} + 1, 2^k + 1)$. **[IH]**
**WTP:** $P((3x - 2y, x))$, i.e., there exists $k'$ such that $(3x - 2y, x) = (2^{k'+1} + 1, 2^{k'} + 1)$.

$$\begin{aligned} 3x - 2y &= 3(2^{k+1} + 1) - 2(2^k + 1) && \text{\# By IH} \\ &= 3 * 2^{k+1} + 3 - 2^{k+1} - 2 \\ &= 2 * 2^{k+1} + 1 \\ &= 2^{k+2} + 1 \end{aligned}$$

Let $k' = k + 1$. Then $k' \in \mathbb{N}$.
Then $3x - 2y = 2^{k'+1} + 1$.
Also, by IH, $x = 2^{k+1} + 1$. So, $x = 2^{k'} + 1$.
Then, there exists $k' \in \mathbb{N}$ such that $(3x - 2y, x) = (2^{k'+1} + 1, 2^{k'} + 1)$.
Thus $P((3x - 2y, x))$.

3. Let $G$ be a set defined as follows:

- if $x$ is a propositional variable, then $x \in G$;
- if $f_1, f_2 \in G$, then $\neg f_1 \in G$, and $(f_1 \wedge f_2) \in G$;
- nothing else belongs to $G$.

For a formula $f \in G$, let $c_{not}(f)$ be the number of occurrences of $\neg$ in $f$, and $c_{and}(f)$ be the number of occurrences of $\wedge$ in $f$. Let $H = \{f \in G : c_{not}(f) = c_{and}(f)\}$. That is, $H$ is the set of formulas in $G$ with equal number of $\neg$'s and $\wedge$'s.
Prove that for any formula $f \in G$, there is a formula $f'$ such that $f' \in H$ and $f'$ and $f$ are logically equivalent.

**Solution:** $P(f)$ : There exists $f'$ such that $f' \in H$ and $f'$ and $f$ are logically equivalent.
We use structural induction (on $G$) to prove that for all $f \in G$, $P(f)$ holds.

**Base Case:** For $f = x$, where $x$ is a propositional variable, let $f' = x$.
Then $f'$ uses no connectives at all. So $f' \in G$ and $c_{not}(f') = 0 = c_{and}(f')$. Thus $f' \in H$.
Also, $f'$ and $f$ are logically equivalent. Therefore $P(f)$ holds.

**Induction Step:** Assume $f_1, f_2 \in G$. By definition, $\neg f_1 \in G$, and $(f_1 \wedge f_2) \in G$.
Suppose $P(f_1)$ and $P(f_2)$, i.e., there exist $f_1', f_2'$ such that $f_1', f_2' \in H$ and $f_1'$ is logically equivalent to $f_1$, and $f_2'$ is logically equivalent to $f_2$. **IH**
**WTP:** $P(f)$ holds for (A) $f = \neg f_1$ and (B) $f = (f_1 \wedge f_2)$.

Case (A): For $f = \neg f_1$, let $f' = \neg(f_1' \wedge f_1')$. Then

$$\begin{aligned} c_{not}(f') &= 1 + 2 * c_{not}(f_1') \\ &= 1 + 2 * c_{and}(f_1') && \text{\# By IH, } c_{not}(f_1') = c_{and}(f_1') \\ &= c_{and}(f'). \end{aligned}$$

By the IH, $f_1' \in G$, and so, considering the inductive rules, $\neg(f_1' \wedge f_1') \in G$.
By the IH, $\neg(f_1' \wedge f_1')$ is logically equivalent to $\neg(f_1 \wedge f_1)$, which is logically equivalent to $\neg f_1$, which by assumption is $f$.
Therefore $P(f)$.

Case (B): For $f = (f_1 \wedge f_2)$, let $f' = (\neg\neg f_1' \wedge (f_2' \wedge f_2'))$. Then

$$\begin{aligned} c_{not}(f') &= 2 + c_{not}(f_1') + 2 * c_{not}(f_2') \\ &= 2 + c_{and}(f_1') + 2 * c_{and}(f_2') \qquad \text{\# By IH, } c_{not}(f_1') = c_{and}(f_1') \text{ and } c_{not}(f_2') = c_{and}(f_2') \\ &= c_{and}(f'). \end{aligned}$$

By the IH, $f_1', f_2' \in G$, and so, considering the inductive rules, $(\neg\neg f_1' \wedge (f_2' \wedge f_2')) \in G$.
By the IH, $(\neg\neg f_1' \wedge (f_2' \wedge f_2'))$ is logically equivalent to $(\neg\neg f_1 \wedge (f_2 \wedge f_2))$, which is logically equivalent to $(f_1 \wedge (f_2 \wedge f_2))$, which is logically equivalent to $(f_1 \wedge f_2)$, which by assumption is $f$.
Therefore $P(f)$.