CS 2742 (Logic in Computer Science) – Winter 2014
Lecture 22

Antonina Kolokolova

March 19, 2014

6.3 Equivalences of well-ordering, induction and complete induction.

Theorem 1. *Well-ordering principle, (weak) induction and strong induction are all equivalent to each other.*

Here is a very brief (and technical) outline of the main structure of the proof of the equivalences. The structure of the proof is circular: first we show that well-ordering implies induction, then that induction implies strong induction, and finally strong induction implies well-ordering, completing the cycle of implications.

Proof.

1) Well-ordering implies induction.
   Assume well-ordering holds. Let $0 \in A$ and let $\forall i \in \mathbb{N}$, if $i \in A$ then $i+1 \in A$. Need to show $\mathbb{N} \subset A$. The rest is by contradiction. Look at $\bar{A} = \mathbb{N} - A$. If $\mathbb{N} \not\subseteq A$, then $\bar{A}$ is nonempty. By well-ordering, $\bar{A}$ has a minimal element j. That element is > 0, since $0 \in A$. Then $j-1$ is a natural number. But then $(j-1)+1$ must be in A. Contradiction.

2) Induction implies complete induction.
   Prove by induction the following property: $P'(n) = \forall i < nP(i)$.

3) Complete induction implies well-ordering.
   Let A be a subset of $\mathbb{N}$ with no minimal element. Show that A is empty. For any $i \in \mathbb{N}$, any number less than i is not in A. Then $i \notin A$ either (it would be the minimal element of A then). Look at the complement of A, $\bar{A}$. By complete induction, if any natural number less than i is in $\bar{A}$, then i is also in $\bar{A}$. But then every natural number is in $\bar{A}$. So A is empty.

□

6.4 Recursive definitions

Definition 1. *A recursive definition consists of:*

1) ***Base of recursion****: a statement that certain objects belong to a set.*

2) ***Recursion:*** *a collections of rules indicating how to form new set objects from those already known to be in the set.*

3) ***Restriction****: A statement that no objects belong to the set other than those coming from the base and the recursion rules.*

Example 1. Fibonacci: $F_0 = F_1 = 1$, and $F_k = F_{k-1} + F_{k-2}$.

Example 2. (Propositional formulas.)
Here we will give a formal definition of formulas of propositional logic.
Base of the recursion: propositional variables $p, q, r, \dots$ and constants F, T are propositional formulas.
Recursion: If ϕ and ψ are propositional formulas, so are $\neg\phi$, $\phi \vee \psi$, $\phi \wedge \psi$, $\phi \to \psi$ and $\phi \leftrightarrow \psi$, as $(\neg\phi)$, $(\phi \vee \psi)$ and so on.
Restriction: ... and nothing else is a propositional formula.
For example, $(p \vee \neg q) \wedge T$ is a propositional formula, because it is made out of a $\wedge$ of $(p \vee \neg q)$ and T, and both of them are propositional formulas: T because it satisfies the base of induction, and $(p \vee \neg q)$ because it is a $\vee$ of two formulas p and $\neg q$, the first of which again satisfies the base case, and the second is a $\neg$ of a formula which is a base case.

Example 3. (Arithmetic expressions)
Base of the recursion: rational numbers and variables $x, y, z, \dots$ are arithmetic expressions.
Recursion: For any two arithmetic expressions A and B, $A+B$, $A-B$, $A*B$, A/B, $(A+B)$, $(A-B)$, $(A*B)$, (A/B) are arithmetic expressions.
Restriction:... and nothing else.
For example, $3 + 5 * x$ is an arithmetic expression.