CS 2742 (Logic in Computer Science) – Fall 2008
Lecture 24

Antonina Kolokolova

November 25, 2008

7.1 Induction

$$\text{for all predicates } P, (P(0) \wedge \forall k(P(k) \rightarrow P(k+1))) \rightarrow \forall n P(n)$$

Structure of a proof by induction:

1) **Predicate** State which $P(n)$ you are proving as a function of n.

2) **Base case:** Prove $P(a)$.

3) **Induction hypothesis:** State "Assume $P(k)$ holds" explicitly.

4) **Induction step:** Show how $P(k) \rightarrow P(k+1)$. That is, assuming $P(k)$ derive $P(k+1)$.

Example 1. Recall the following question: show that every amount of change ≥ 8 can be paid with only 3c and 5c coins. This time we will prove this using induction
Let $P(n) : \exists i, j \geq 0 \ n = 3i + 5j$
Base case: Let $n = 8$. Then $n = 3 + 5$, $i = j = 1$. For this method of solving the problem, it is also convenient to have a base case $n = 9 = 3 \cdot 3$, $i = 3$, $j = 0$.
Induction hypothesis: Assume that $\exists i, j \geq 0 \ k = 3i + 5j$. This assumption gives us the i and j which we will be using in the induction step.
Induction step: We want to show that $\exists i', j' \geq 0$ such that $k + 1 = 3i' + 5j'$. Look at i and j given to us by induction hypothesis, that is, i and j such that $k = 3i + 5j$. Consider the following two cases.
Case 1: $j > 0$. That is, at least one 5c coin was used to make k. Then we can replace this 5c coin with two 3c coins to get $k + 1$. That is, $i' = i + 2$ and $j' = j' - 1$, so $k + 1 = 3i' + 5j' = 3(i + 2) + 5(j - 1)$.
Case 2: $j = 0$. Suppose that there was no 5c coin used to make up k, that is, $k = 3i$ for

some i. Since $k \geq 8$, $i \geq 3$. Now, to make $k+1$ we can take three 3c coins out of i used to make up k and replace them by two 5c coins. That is, $i' = i - 3$ and $j' = 2$. Since $i \geq 3$, $i' \geq 0$, and $k+1 = 3i' + 5j'$. This completes the proof.

Note how here we actively used the values i and j, existence of which was given to us by the induction hypothesis, to build our new i and j existence of which we were proving in the induction step. This is one reason why it is good to write out the induction hypothesis: to see the values that are available to be used in the induction step.

Example 2. Here is an example of proving an unequality by induction: $n^2 \leq 2^n$ for $n > 3$. You have seen this already in the assignment: this unequality says that for large enough numbers the size of a powerset 2^A is always larger than the size of a Cartesian product of a set with itself $A \times A$. Another way of looking at this unequality is from the algorithmic point of view: it says that for large enough input size n, an algorithm that runs in time $O(n^2)$ always runs faster than an algorithm that runs in time $O(2^n)$ and is not in $O(n^2)$.

We will prove this unequality by induction.
Predicate $P(n): \ n^2 \leq 2^n$.
Base case: $P(4): 4^2 = 16 \leq 2^4$
Induction hypothesis: assume that for $k > 3$, $k^2 \leq 2^k$.
Induction step: Assuming $P(k)$, prove that $(k+1)^2 \leq 2^{k+1}$.
First, $(k+1)^2 = k^2 + 2k + 1$ and $2^{k+1} = 2 \cdot 2^k = 2^k + 2^k$. By induction hypothesis, $k^2 + 2k + 1 \leq 2^k + 2k + 1$. It remains to show that $2k + 1 \leq 2^k$, where this is the second "copy" of 2^k in $2^k + 2^k$ expression. We could prove this by doing another induction proof, but in this case it can be done easier. Notice that it is sufficient to show that $2k + 1 \leq k^2$, because then by induction hypothesis we will get $2k+1 \leq k^2 \leq 2^k$. To see that $2k+1 \leq k^2$, divide both sides of the unequality by k. Since k is positive, this preserves the unequality, resulting in $2 + 1/k \leq k$. But we do know that $k > 3$, so $2 + 1/k < 3 < k$. Therefore, $2k + 1 \leq k^2 \leq 2^k$, and so $k^2 + (2k+1) \leq 2^k + 2^k$, completing the proof.