CS70, Fall 2003 **Discussion #3** **Amir Kamil**
UC Berkeley **9/12/03**

Topics: Bad Proofs, Generalized Induction, Recursion Proofs, Well-Ordering

1 Bad Proofs

Consider the following proof:

Claim: $\forall x \in .\, \mathbf{N} \forall y \in \mathbf{N}.\ x \cdot y = 0$.
Proof by induction over pairs.
Let $P(x, y) =$ “$x \cdot y = 0''$.

- Base case: $P(0, 0)$ is trivially true.
- Inductive step: $(\forall x', y'.x' + y' < x + y \implies P(x', y')) \implies P(x, y)$
  Consider the pair $(x - 1, y)$. By the inductive inductive hypothesis, $(x - 1) \cdot y = 0$. Thus $x = 1$ or $y = 0$. Now consider $(x, y - 1)$. By the inductive hypothesis, $x \cdot (y - 1) = 0$, so $x = 0$ or $y = 1$. Finally, consider $(x, y - 2)$. By the inductive hypothesis, $x \cdot (y - 2) = 0$, so $x = 0$ or $y = 2$. The only set of values that solves the above three conditions is $x = 0$, $y = 0$. Thus $x \cdot y = 0$.

What is wrong with the above proof? It is the application of the inductive hypothesis. It is not necessarily true that $x - 1$, $y - 1$, or $y - 2$ are naturals, so in some cases the application is illegal. This error occurs more frequently in induction over pairs than over natural numbers, so it is something to watch out for.

2 Generalized Induction

The generalized induction principle for a property P, a set X, and a well-founded[1] relation $\prec$ on X is:

- Base cases: $P(x)$ for each $x \in X$ that is minimal with respect to $\prec$.
- Inductive step: if $P(y)$ is true for all $y \prec x'$, then $P(x')$.

If the base cases and the inductive step hold, then $P(x)$ for all $x \in X$.

Consider, for example, the following function on the pairs of natural numbers $\mathbf{N}^2$:

```
s(x, y):
  if (x == 0 && y == 0) return 0;
  else if (y == 0) return s(x-1, y) + 1;
  else return s(x, y-1) + y;
```

I claim that this function always returns $y \cdot (y + 1)/2 + x$ for all naturals x and y. We can prove this fact using generalized induction over pairs.

First, it is necessary to define the relation $\prec$ on pairs of natural numbers. There are different well-founded relations that work, but perhaps the easiest is the following:

Given two pairs (x, y) *and* (x', y') *in* $\mathbf{N}^2$, $(x, y) \prec (x', y')$ *iff* $(x + y) < (x' + y')$.

[1] Make sure you understand the difference between *well-founded* and *well-ordered*. Well-founded applies to relations with respect to a set (i.e. a relation $\prec$ on a set $\mathbf{S}$ is well-founded), while well-ordered applies to a set with respect to a relation (i.e. a set $\mathbf{S}$ is well-ordered by a relation $\prec$).

Now we can prove our claim using generalized induction over pairs. Define $P(x, y) =$ "`s(x, y)` $= y \cdot (y + 1)/2 + x$."

Theorem 3.1: $\forall x \in \mathbf{N}.\ \forall y \in \mathbf{N}.$ `s(x, y)` $= y(y + 1)/2 + x$.
Proof by induction over pairs:

- Base case: $P(0, 0)$
  Since `x == 0` and `y == 0`, `s(0, 0)` $= 0 \cdot (0 + 1)/2 + 0 = 0$.

- Inductive step: $(\forall x', y'. x' + y' < x + y \implies P(x', y')) \implies P(x, y)$
  As suggested by the structure of the function `s()`, there are two cases:

  1. Case 1: $y = 0$
     Then we have

$$\begin{aligned} \texttt{s(x, y)} &= 1 + \texttt{s(x-1, y)} && \text{(since } \texttt{x != 0} \text{ and } \texttt{y == 0}\text{)} \\ &= 1 + y(y + 1)/2 + x - 1 && \text{(by the inductive hypothesis)} \\ &= y(y + 1)/2 + x. && () \end{aligned}$$

  2. Case 2: $y \neq 0$
     Then we have

$$\begin{aligned} \texttt{s(x, y)} &= \texttt{s(x, y-1)} + y && \text{(since } \texttt{x != 0} \text{ and } \texttt{y != 0}\text{)} \\ &= y(y - 1)/2 + x + y && \text{(by the inductive hypothesis)} \\ &= (y^2 - y + 2y)/2 + x && () \\ &= y(y + 1)/2 + x. && () \end{aligned}$$

  In both cases, we have `s(x, y)` $= y(y + 1)/2 + x$.

Now we really want to make sure that we don't repeat the error in the bad proof above by establishing that our application of the inductive hypothesis is valid. We see that we only apply the inductive hypothesis on $(x - 1, y)$ when $y = 0$ but $x \neq 0$ (since otherwise it would have fallen into the base case), and on $(x, y - 1)$ when $y \neq 0$. Thus our application of the inductive hypothesis is indeed valid.

3 Recursion Proofs

Induction and recursion are very closely related. Both have base cases, and while induction has the inductive step, recursion has the recursive step. Both reformulate a problem in terms of smaller ones, which are already known to work correctly. Thus it is only natural to use induction in order to prove facts about recursive functions.

Consider the set of strings over the English alphabet. A string over an alphabet Σ is a sequence of letters $a_1 a_2 \cdots a_n$ such that each $a_i \in \Sigma$. The length of the preceding string is n. There also exists the empty string λ, whose length is defined to be 0. Now consider the `len()` function:

```
len(a = $a_1 a_2 \cdots a_n$):
  if (a == $\lambda$) return 0;
  else return 1 + len($a_2 \cdots a_n$);
```

This function attempts to compute the length of a string. In order to prove that it works correctly, we use induction.

The first step, as always, is to write a proposition. Now we have two choices here. We can do a normal induction over the naturals, or we can do a generalized induction over strings. Let's do the former. Define

$$P(n) = \text{“}\texttt{len(s)} \text{ correctly computes the length for all length } n \text{ strings } \texttt{s}.\text{”}$$

Now our claim is that $P(n)$ is true for all natural numbers n.

Theorem 3.2: $\forall n \in N$. `len(s)` correctly computes the length for all length n strings `s`.
Proof by induction:

- Base case: $P(0)$
  `len(`λ`)` $= 0$ is the correct length of λ.

- Inductive step: $P(n) \implies P(n+1)$
  Consider an arbitrary string of a length $n+1$. $a = a_1 \cdots a_{n+1}$ by definition of length. Now `len(a)` returns 1+`len(`$a_2 \cdots a_{n+1}$`)`. By the inductive hypothesis, `len(`$a_2 \cdots a_{n+1}$`)`$= n$, the length of $a_2 \cdots a_{n+1}$. Thus `len(a)` returns $1 + n = n + 1$, the correct length of a.

Note that we have glossed over some of the issues that would have to be dealt with when proving that a function in some real computer program works. For example, what happens when the input to `len()` has a longer length than the size of the maximum 32 bit integer? We also would have to appeal to a set of axioms regarding the operational semantics of the particular language the function is written in, to prove that it picks the correct branch in the conditional statement. But for now, we will ignore such concerns.

4 Well-Ordering

Besides being used in induction, well-ordering can be used directly in proving claims. The general scheme for a proof by well-ordering to show that a claim $P(x)$ is true for all x in some set $\mathbf{X}$ is to first assume that there are some elements for which it is false. Call the set of such elements $\mathbf{X}'$. Then take the minimal element in $\mathbf{X}'$ with respect to some well-founded relation $\prec$, and show that it cannot possiblty be the minimal element in $\mathbf{X}'$. This is a contradiction, so $P(x)$ must be true for all $x \in \mathbf{X}$.

As an example, let's prove that all natural numbers greater than 1 can be written as the product of primes. Let $P(n) =$ “n can be written as a product of primes.” Then we claim that for all naturals $n > 1$, $P(n)$.

Theorem 3.3: $\forall n \in \mathbf{N}.\ n > 1 \implies n$ can be written as a product of primes.
Proof: Suppose that $P(n)$ is false for some naturals greater than 1. Let $\mathbf{S}$ be the set of all such elements. Then by the well-ordering principle, $\mathbf{S}$ has a minimal element, so let m be that element. Now m cannot be prime, otherwise it can be written as a product of itself, resulting in a contradiction. Thus by definition of prime, m must have a positive divisor other than 1 and m. Let d be its smallest such divisor. d must be prime, otherwise it has a divisor other than 1 or d, which also divides m, contradicting our choice of d. Now consider $\frac{m}{d}$, which must be an integer by definition of divisor, and must be positive since m and d are. There are two cases:

1. Case 1: $\frac{m}{d}$ can be written as the product of primes $p_1 \cdot p_2 \cdot \ldots \cdot p_k$. Then m can be written as $p_1 \cdot p_2 \cdot \ldots p_k \cdot d$, i.e. as a product of primes, which is a contradiction.

2. Case 2: $\frac{m}{d}$ cannot be written as a product of primes. But since d is greater than 1, $\frac{m}{d}$ is smaller than m, contradicting our choice of m.

Since either case results in a contradiction, it must be the case that m does not exist, i.e. $\mathbf{S}$ is empty. Thus $P(n)$ is true for all naturals $n > 1$.

Notice the reference to *the principle of well-ordering*. This principle is the fact that any set of natural numbers must have a minimal element with respect to the relation $<$.

The above proof actually applies the well-ordering principle twice. Can you find its second application? Is it valid to apply it in this case?