CS 70 Fall 2006	Discrete Mathematics for CS Papadimitriou & Vazirani	Lecture 3

Induction

Induction is an extremeley powerful tool in mathematics. It is a way of proving propositions that hold for all natural numbers (i.e., universally quantified $\forall k \in \mathbf{N}$):

1) $\forall k \in \mathbf{N}$, $0+1+2+3+\cdots+k = \frac{k(k+1)}{2}$

2) $\forall k \in \mathbf{N}$, the sum of the first k odd numbers is k^2.

3) Any graph with k vertices and k edges contains a cycle.

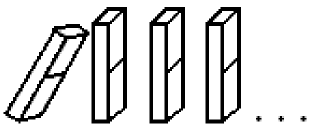
Each of these propositions is of the form $\forall k \in \mathbf{N}\ P(k)$. For example, in the first proposition, $P(k)$ is the statement $0+1+\cdots+k = \frac{k(k+1)}{2}$, $P(0)$ says $0 = \frac{0(0+1)}{2}$, $P(1)$ says $0+1 = \frac{1(1+1)}{2}$, etc. The *principle of induction* asserts that you can prove $P(k)$ is true $\forall k \in \mathbf{N}$, by following these three steps:

Base Case: Prove that $P(0)$ is true.

Inductive Hypothesis: Assume that $P(k)$ is true.

Inductive Step: Prove that $P(k+1)$ is true.

The principle of induction formally says that if $P(0)$ and $\forall n \in \mathbf{N}\ (P(n) \Longrightarrow P(n+1))$, then $\forall n \in \mathbf{N}\ P(n)$. Intuitively, the base case says that $P(0)$ holds, while the inductive step says that $P(0) \Longrightarrow P(1)$, and $P(1) \Longrightarrow P(2)$, and so on. The fact that this domino effect eventually shows that $\forall n \in \mathbf{N}\ P(n)$ is what the principle of induction (or the induction axiom) states. In fact, dominoes are a wonderful analogy: we have a domino for each proposition $P(k)$. The dominoes are lined up so that if the k^{th} domino is knocked over, then it in turn knocks over the $k+1^{st}$. Knocking over the k^{th} domino corresponds to proving $P(k)$ is true. So the induction step corresponds to the fact that the k^{th} domino knocks over the $k+1^{st}$ domino. Now, if we knock over the first domino (the one numbered 0), then this sets off a chain reaction that knocks down all the dominoes.



Theorem: $\forall k \in \mathbf{N}$, $\displaystyle\sum_{i=0}^{k} i = \frac{k(k+1)}{2}$.

Proof (by induction on k):

- Base Case: $P(0)$ asserts: $\displaystyle\sum_{i=0}^{0} i = \frac{0(0+1)}{2}$. This clearly holds, since the left and right hand sides both equal 0.

- Inductive Hypothesis: Assume $P(k)$ is true. That is, $\sum_{i=0}^{k} i = \frac{k(k+1)}{2}$.
- Inductive Step: We must show $P(k+1)$. That is, $\sum_{i=0}^{k+1} i = \frac{(k+1)(k+2)}{2}$:

$$\begin{aligned}
\sum_{i=0}^{k+1} i &= (\sum_{i=0}^{k} i) + (k+1) \\
&= \frac{k(k+1)}{2} + (k+1) && \text{(by the inductive hypothesis)} \\
&= (k+1)(\frac{k}{2}+1) \\
&= \frac{(k+1)(k+2)}{2}
\end{aligned}$$

Note the structure of the inductive step. You try to show $P(k+1)$ with the assumption that $P(k)$ is true. The idea is that $P(k+1)$ by itself is a difficult proposition to prove. Many difficult problems in computer science are solved by breaking the problem into smaller, easier ones. This is precisely what we did in the inductive step - $P(k+1)$ is difficult to prove, but we were able to recursively define it in terms of $P(k)$.

We will now look at another proof by induction, but first we will introduce some notation and a definition for divisibility. We write $a|b$ (a divides b, or b is divisible by a), and define divisibility as follows: for all integers a and b, $a|b$ if and only if for some integer q, $b = aq$.

Theorem: $\forall n \in \mathbf{N}$, $n^3 - n$ is divisible by 3.

Proof (by induction over n):

- Base Case: $P(0)$ asserts that $3|(0^3 - 0)$ or $3|0$, which is clearly true (since $0 = 3 \cdot 0$).
- Inductive Hypothesis: Assume $P(n)$ is true. That is, $3|(n^3 - n)$.
- Inductive Step: We must show that $P(n+1)$ is true, which asserts that $3|((n+1)^3 - (n+1))$. Let us expand this out:

$$\begin{aligned}
(n+1)^3 - (n+1) &= n^3 + 3n^2 + 3n + 1 - (n+1) \\
&= (n^3 - n) + 3n^2 + 3n \\
&= 3q + 3(n^2+n),\ q \in \mathbf{Z} && \text{(by the inductive hypothesis)} \\
&= 3(q + n^2 + n)
\end{aligned}$$

Hence, by the principle of induction, $\forall n \in \mathbf{N}$, $3|(n^3 - n)$.

The next example we will look at is an inequality between two functions of n. Such inequalities are useful in computer science when showing that one algorithm is more efficient than another. Notice that for this example, our base case does not begin at $n = 0$.

Theorem: $\forall n \in \mathbf{N}$, $n > 1 \Longrightarrow n! < n^n$.

Proof (by induction over n):

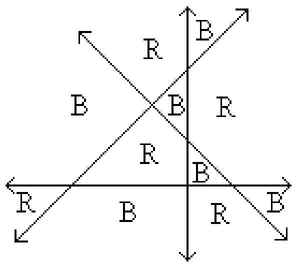
- Base Case: $P(2)$ asserts that $2! < 2^2$, or $2 < 4$, which is clearly true.
- Inductive Hypothesis: Assume $P(n)$ is true (i.e., $n! < n^n$).
- Inductive Step: We must show $P(n+1)$, which states that $(n+1)! < (n+1)^{n+1}$. Let us begin with the left side of the inequality:

$$\begin{aligned}(n+1)! &= (n+1)\cdot n! \\ &< (n+1)\cdot n^n \qquad \text{(by the inductive hypothesis)} \\ &< (n+1)\cdot(n+1)^n \\ &= (n+1)^{n+1}\end{aligned}$$

Hence, by the induction principle, $\forall n \in \mathbf{N}$, if $n > 1$, then $n! < n^n$.

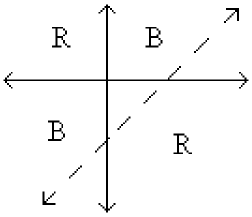
Induction is a widely applicable proof technique. We now turn to a geometrical example:

Two Color Theorem: There is a famous theorem called the four color theorem. It states that any map can be colored with four colors such that any two adjacent countries (which share a border) must have different colors. The four color theorem is very difficult to prove, and a proof was claimed in 1969 but turned out to be fallacious. It was not until 1976 that the theorem was finally proven with the aid of a computer. We consider a simpler scenario, where we divide the plane into regions by drawing straight lines. We want to know if we can color this map using no more than two colors (say red and blue) such that no two regions that share a boundary have the same color. Here is an example of a two-colored map:



We will prove this "two color theorem" by induction on n, the number of lines:

- Base Case: Prove that $P(0)$ is true, which is the proposition that a map with $n = 0$ lines can be can be colored using no more than two colors. But this is easy, since we can just color the entire plane using one color.
- Inductive Hypothesis: Assume $P(n)$. That is, a map with n lines can be two-colored.
- Inductive Step: Prove $P(n+1)$. We are given a map with $n+1$ lines and wish to show that it can be two-colored. Let's see what happens if we remove a line. With only n lines on the plane, we know we can two-color the map (by the inductive hypothesis). Let us make the following observation: if we swap red $\leftrightarrow$ blue, we still have a two-coloring. With this in mind, let us place back the line we removed, and leave colors on one side of the line unchanged. On the other side of the line, swap red $\leftrightarrow$ blue. We claim that this is a valid two-coloring for the map with $n+1$ lines.



Why does this work? We can say with certainty that regions which do not border the line are properly two-colored. But what about regions that do share a border with the line? We must be certain that any two such regions have opposite coloring. But any two regions that border the line must have been the same region when the line was removed, so the reversal of color on one side of the line guarantees an opposite coloring.

Now that we know how to prove propositions by induction, let us try to prove the following amazing fact.

Theorem: All horses are the same color.

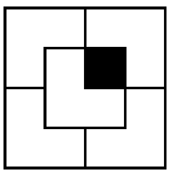
Proof (by induction on the number of horses):

- Base Case: $P(1)$ is certainly true, since with just one horse, all horses have the same color.
- Inductive Hypothesis: Assume $P(n)$, which is the statement that n horses all have the same color.
- Inductive Step: Given a set of $n+1$ horses, we can exclude the last horse in the set and apply the inductive hypothesis to the first n horses, showing that they all have the same color. By a similar argument, we can conclude that the last n horses also have the same color. It follows that all $n+1$ horses have the same color. Thus, by the principle of induction, all horses have the same color.

Clearly, it is not true that all horses are of the same color, so where did we go wrong in our proof by induction? We seemed to show the base case and the inductive step without any problems. It turns out that the flaw is relatively difficult to diagnose, since the inductive step is true for the "typical" case where, say, $n = 3$. However, when we look at the root of the problem, the error comes up in the inductive step when $n = 1$. In this case, our proof breaks down since the first horse and the last horse have no horses in common, and for this reason the two horses may not have the same color.

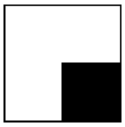
Strengthening the Inductive Hypothesis

Imagine that we are given L-shaped tiles (i.e., a 2×2 square tile with a missing 1×1 square), and we want to know if we can tile a $2^n \times 2^n$ courtyard with a missing 1×1 square in the middle. Here is an example of a successful trial in the case that $n = 2$:



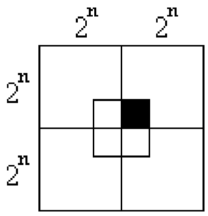
Let us try to prove the proposition by induction on n.

- Base Case: Prove $P(1)$. This is the proposition that a 2×2 courtyard can be tiled with L-shaped tiles with a missing 1×1 square in the middle. But this is easy:

- Inductive Hypothesis: Assume $P(n)$ is true. That is, we can tile a $2^n \times 2^n$ courtyard with a missing 1×1 square in the middle.

- Inductive Step: We want to show that we can tile a $2^{n+1} \times 2^{n+1}$ courtyard with a missing 1×1 square in the middle. Let's try to reduce this problem so we can apply our inductive hypothesis. A $2^{n+1} \times 2^{n+1}$ courtyard can be broken up into four smaller courtyards of size $2^n \times 2^n$, each with a missing 1×1 square as follows:

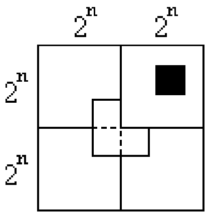
  

  But the holes are not in the middle of each $2^n \times 2^n$ courtyard, so the induction hypothesis does not help! How should we proceed? We should strengthen our inductive hypothesis!

What we are about to do is completely counter-intuitive. It's like attempting to lift 100 pounds, failing, and then saying "I couldn't lift 100 pounds. Let me try to lift 200," and then succeeding! Instead of proving that we can tile a $2^n \times 2^n$ courtyard with a hole in the middle, we will try to prove something stronger: that we can tile the courtyard with the hole being *anywhere*. It is a trade-off: we have to prove more, but we also get to assume a stronger hypothesis. The base case is the same - so we will work on the inductive hypothesis and step.

- Inductive Hypothesis (second attempt): Assume $P(n)$ is true, which states that we can tile a $2^n \times 2^n$ courtyard with a missing 1×1 square anywhere.

- Inductive Step (second attempt): As before, we can break up the $2^{n+1} \times 2^{n+1}$ courtyard as follows.

  By placing a tile as shown, we get four $2^n \times 2^n$ courtyards, each with a 1×1 hole. The stronger induction hypothesis applies, and each of the four courtyards can be successfully tiled. Thus, we have proven that we can tile a $2^{n+1} \times 2^{n+1}$ courtyard with a hole anywhere!

Strong Induction

Strong induction is very similar to simple induction, with the exception of the inductive hypothesis. With strong induction, instead of just assuming $P(k)$ is true, you assume the stronger statement that $P(0)$, $P(1)$, ..., and $P(k)$ are all true (i.e., $P(0) \wedge P(1) \wedge \cdots \wedge P(k)$ is true or in more compact notation $\wedge_{i=0}^{k} P(i)$ is true). Strong induction sometimes makes the proof of the inductive step much easier since we get to assume a stronger statement, as illustrated in the next example.

Theorem: Every natural number $n > 1$ can be written as a product of primes.

Recall that a number n is prime if 1 and n are its only divisors. Let $P(n)$ be the proposition that n can be written as a product of primes. We will prove that $P(n)$ is true for all $n \geq 2$.

- Base Case: We start at $n = 2$. This holds, since 2 is a prime number.
- Inductive Hypothesis: Assume $P(k)$ is true for $2 \leq k \leq n$: i.e., every number $k : 2 \leq k \leq n$ can be written as a product of primes.
- Inductive Step: We must show that $n+1$ can be written as a product of primes. We have two cases - either $n+1$ is a prime number, or it is not. For the first case, if $n+1$ is a prime number, then we are done. For the second case, if $n+1$ is not a prime number, then by definition $n+1 = xy$, where x,$y \in \mathbf{Z}^{+}$ and $1 < x, y < n+1$. By the inductive hypothesis, x and y can be written as a product of primes (since $x, y \leq n$). Therefore, $n+1$ can also be written as a product of primes.

Why does this proof fail if we were to use simple induction? If we only assume $P(n)$ is true, then we cannot apply our inductive hypothesis to x and y. For example, if we were trying to prove $P(42)$, we might write $42 = 6 \times 7$, and then it is useful to know that $P(6)$ and $P(7)$ are true. However, with simple induction, we could only assume that 41 can be written as a product of primes - a fact that is not useful in establishing $P(42)$.

Consider the following example, which is of immense interest to post offices and their customers: any integer amount of postage from 8¢ upwards can be composed from 3¢ and 5¢ stamps. With a strong induction, we can make the connection between $P(n+1)$ and earlier facts in the sequence. In particular, $P(n-2)$ is relevant because $n+1$ can be composed from the solution for $n-2$ plus one 3¢ stamp. So the inductive step works if $P(n-2)$ is known already. This will not be the case when $n+1$ is 9 or 10, so we will need to handle these cases separately.

Proof (by strong induction over n, the number of cents):

- Base Case: Prove $P(8)$, which is the proposition that 8¢ of postage can be composed from 3¢ and 5¢ stamps. This is true, requiring one of each.
- Inductive Hypothesis: Assume $P(m)$ is true for all naturals numbers $8 \leq m \leq n$. That is, m¢ of postage can be composed from 3¢ and 5¢ stamps.
- Inductive step: Prove $P(8) \wedge \cdots \wedge P(n) \implies P(n+1)$. We need to show that (n+1)¢ of postage can be composed from 3¢ and 5¢ stamps. First, the cases where $n+1$ is 9 or 10 must be proved separately. $P(9)$ is true, since 9¢ can be composed from three 3¢ stamps. $P(10)$ is clearly true also, since 10¢ can be composed from two 5¢ stamps. For all natural numbers $n+1 > 10$, the inductive hypothesis entails the proposition $P(n-2)$. If $(n-2)$¢ can be composed from 3¢ and 5¢ stamps, then $(n+1)$¢ can be composed from 3¢ and 5¢ stamps simply by adding one more 3¢ stamp.

Simple Induction vs. Strong Induction

We have seen that strong induction makes certain proofs easy when simple induction seems to fail. A natural question to ask then, is whether the strong induction axiom is logically stronger than the simple induction axiom. In fact, the two methods of induction are logically equivalent. Clearly anything that can be proven by simple induction can also be proven by strong induction (convince yourself of this!). Suppose we can prove by strong induction that $\forall n\ P(n)$. Let $Q(k) = P(0) \wedge \cdots \wedge P(k)$. Let us prove $Q(k)$ by simple induction. The proof is modeled after the strong induction proof of $\forall n\ P(n)$. That is, we want to show $P(0)\wedge\cdots\wedge P(k) \Rightarrow P(0)\wedge\cdots\wedge P(k)\wedge P(k+1)$. But this is true iff $P(0) \wedge \cdots \wedge P(k) \Rightarrow P(k+1)$. This is exactly what the strong induction proof of $\forall n\ P(n)$ establishes. Therefore, we can establish $\forall n\ Q(n)$ by simple induction.

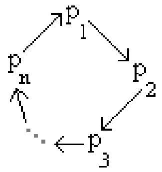
Well Ordering Principle

How can the induction axiom fail to be true? Recall that the axiom says the following: $[P(0) \wedge \forall n\ P(n) \Rightarrow P(n+1)] \Longrightarrow \forall n\ P(n)$. Assume for a contradiction that $\neg(\forall n \in \mathbf{N}\ P(n))$. Then this means that $\exists n : P(n)$ is false. In other words, if we have a collection of propositions $P(0)$, $P(1)$, $\ldots$, $P(n-1)$, and $P(n)$, then some of these propositions must be false. Let m be the smallest such n for which $P(m-1)$ is true and $P(m)$ is false. But this directly contradicts the fact that $P(m-1) \Longrightarrow P(m)$. It may seem as though we just proved the induction axiom. But what we have actually done is to show that the induction axiom follows from another axiom, which was used implicitly in defining "the first m for which $P(m)$ is false."

Well ordering principle: If $S \subseteq \mathbf{N}$ and $S \neq \emptyset$, then S has a smallest element.

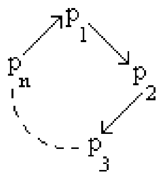
We assumed something when defining m that is usually taken for granted: that we can actually find a smallest number. This cannot always be accomplished, and to see why consider the set $\{x : 0 < x < 1,\ x \in \mathbb{R}\}$. Whatever number is claimed to be the smallest of the set, we can always find a smaller one. Again, the well ordering principle may seem obvious but it should not be taken for granted. It is only because the natural numbers and any subset of the natural numbers are well ordered that we can find a smallest element. Not only does the principle underlie the induction axioms, but it also has direct uses in its own right.

Round robin tournament: Suppose that we have a set of n players $\{p_1, p_2, \ldots, p_n\}$ in a round robin tournament and we have the scenario where p_1 beats p_2, p_2 beats p_3, $\ldots$, and p_n beats p_1. Then we will call this a cycle:

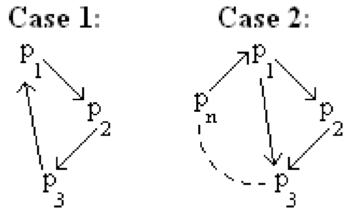


Claim: $\exists$ cycle $\Longrightarrow \exists$ 3-cycle.

Assume for a contradiction that the smallest cycle is:



Let us look at the game between p_1 verus p_3. We have two cases: either p_3 beats p_1, or p_1 beats p_3. In the first case where p_3 beats p_1, then we are done because we have a 3-cycle. In the second case where p_1 beats p_3, we have a shorter cycle and thus, a contradiction. Therefore, if there exists a cycle, then there must exist a 3-cycle as well.



Induction and Recursion

There is an intimate connection between induction and recursion. A recursive definition of a function over the natural numbers specifies the value of the function at small values of n, and defines the value of $f(n)$ for a general n in terms of the value of $f(m)$ for $m < n$. Let us consider the example of the Fibonacci numbers, defined in a puzzle by Fibonacci (in the year 1202):

Fibonacci's puzzle: A certain man put a pair of rabbits in a place surrounded on all sides by a wall. How many pairs of rabbits can be produced from that pair i n a year if it is supposed that every month each pair begets a new pair which fr om the second month on becomes productive?

The initial conditions specified in the puzzle says that the number of pairs of rabbits $F(n)$ in month n satisfies the initial conditions $F(0) = 0$ and when the pair of rabbits is introduced, $F(1) = 1$. Also $F(2) = 1$, since the pair is not yet productive. In month 3, according to the conditions, the pair of rabbits begets a new pair. So $F(3) = 2$. In the fourth month, this new pair is not yet productive, but the original pair is, so $F(4) = 3$. What about $F(n)$ for a general value of n. This is a little tricky to figure out unless you look at it the right way. The number of pairs in month $n-1$ were $F(n-1)$. Of these how many were productive? Only those that were alive in the previous month - i.e. $F(n-2)$ of them. Thus there are $F(n-2)$ new pairs in the n-th month, and so $F(n) = F(n-1) + F(n-2)$. This completes the recursive definition of $F(n)$:

- $F(0) = 0$, and $F(1) = 1$
- For $n \geq 2$, $F(n) = F(n-1) + F(n-2)$

This admittedly simple model of population growth nevertheless illustrates a fundamental principle. Left unchecked, populations grow exponentially over time (exercise: can you show, for example, that $F(n) \geq 2^{(n-1)/2}$). Understanding the significance of this unchecked exponential population growth was a key step that led Darwin to formulate his theory of evolution. To quote Darwin: "There is no exception to the rule that every organic being increases at so high a rate, that if not destroyed, the earth would soon be covered by the progeny of a single pair."'

Be sure you understand that a recursive definition is not circular — even though in the above example $F(n)$ is defined in terms of the function F, there is a clear ordering which makes everything well-defined. Here is a recursive program to evaluate $F(n)$:

```
Function F(n)
If n=0 return 1
If n=1 return 1
Else return F(n-1)+Fib(n-2)
```

Can you figure out how long does this program take to compute $F(n)$? This is a very inefficient way to compute the n-th Fibonacci number. A much faster way is to turn this into an iterative algorithm (this should be a familiar example of turning a tail-recursion into an iterative

algorithm):

```
Function F_2(n)
If n=0 return 1
If n=1 return 1
a = 1
b = 1
For k = 2 to n
a = a+b
b = a
rof
return a
```

Can you show by induction that this new function $F_2(n) = F(n)$. How long does this program take to compute $F(n)$?