CS 70 Discrete Mathematics and Probability Theory
Fall 2013 Vazirani Note 1

Induction

Induction is a basic, powerful and widely used proof technique. It is one of the most common techniques for analyzing programs: proving that they correctly compute the desired function on all inputs and analyzing their running time as a function of input length. Let us start with an example of induction.

Consider the statement: for all natural numbers n, $0+1+2+3+\cdots+n = \frac{n(n+1)}{2}$. Using mathematical notation, we can formally write the statement as: $\forall n \in \mathbb{N}, \sum_{i=0}^{n} i = \frac{n(n+1)}{2}$. In plain english, we read this statement as follows: for all n in $\mathbb{N}$ (the set of natural numbers), the sum of i going from $i=0$ to n is $\frac{n(n+1)}{2}$.

How would you prove this statement? Of course you can substitute small values for n and check that it holds for those cases. While this is a very good way to understand the statement, and often helps you gain insight into why the statement might be true, it does not really prove that the statement holds for all natural numbers n. For example, consider the statement $\forall n \in \mathbb{N}, P(n) = n^2 - n + 41$ is a prime number. If you check the first few inputs, you will find that the results are indeed prime. Indeed, you have to get all the way up to $n = 41$ before finding out that the statement is false!

Below we give a proof by induction that $\forall n \in \mathbb{N}, \sum_{i=0}^{n} i = \frac{n(n+1)}{2}$. If you have never seen an induction proof before, it might seem a bit strange to you. Do not worry! We will soon describe the intuitive picture behind induction, as well as describe the general structure of an induction proof.

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

Proof (by induction on n):

- Base Case: $n = 0$: $\sum_{i=0}^{0} i = 0 = \frac{0(0+1)}{2}$. Correct.
- Inductive Hypothesis: Assume that $\sum_{i=0}^{k} i = \frac{k(k+1)}{2}$.

- Inductive Step: Prove that it also holds for $n = (k+1)$, i.e. $\sum_{i=0}^{k+1} i = \frac{(k+1)(k+2)}{2}$:

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

Hence, by the principle of induction, the theorem holds. ♠

Let's step back and look at the general form of such a proof, and also why it makes sense. Let us denote by $P(n)$ the statement $\sum_{i=0}^{n} i = \frac{n(n+1)}{2}$. So we wish to prove that $\forall n \in \mathbb{N}$, $P(n)$. The *principle of induction* asserts that you can prove $P(n)$ is true $\forall n \in \mathbb{N}$, by following these three steps:

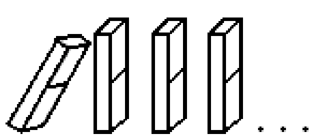
Base Case: Prove that $P(0)$ is true.

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

Inductive Step: Show that it follows that $P(k+1)$ is true.

To understand why induction works, think of the statements $P(n)$ as represented by a sequence of dominoes, numbered from $0,1,2,\ldots,n$, such that $P(0)$ corresponds to the 0^{th} domino, $P(1)$ corresponds to the 1^{st} domino, and so on. 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. And the induction step corresponds to the placement of the dominoes to ensure that if the k^{th} domino falls, in turn it knocks over the $k+1^{st}$ domino. The base case ($n = 0$) knocks over the $0th$ domino, setting off a chain reaction that knocks down all the dominoes.

It is worth examining more closely the induction proof example above. To prove $P(k+1)$, we find within it the statement $P(k)$: $\sum_{i=0}^{k+1} i = (\sum_{i=0}^{k} i) + (k+1)$. This is the key to the induction step.



We will now look at another proof by induction, but first we will introduce some notation and a definition for divisibility. Given integers a and b, we say that a divides b (or b is divisible by a), written as $a|b$, if and only if for some integer q, $b = aq$. In mathematical notation, $\forall a, b \in \mathbb{Z}$, $a|b$ iff $\exists q \in \mathbb{Z} : b = aq$.

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

Proof (by induction over n): Let $P(n)$ denote the statement $\forall n \in \mathbb{N}$, $n^3 - n$ is divisible by 3.

- Base Case: $P(0)$ asserts that $3|(0^3 - 0)$ or $3|0$, which is true since non-zero integer divides 0. (In this case, $0 = 3 \cdot 0$).
- Inductive Hypothesis: Assume $P(k)$ is true. That is, $3|(k^3 - k)$, or $\exists q \in \mathbb{Z}, k^3 - k = 3q$.
- Inductive Step: We must show that $P(k+1)$ is true, which asserts that $3|((k+1)^3 - (k+1))$. Let us expand this out:

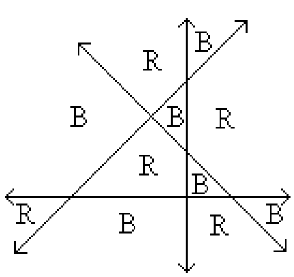
$$\begin{aligned}(k+1)^3 - (k+1) &= k^3 + 3k^2 + 3k + 1 - (k+1) \\ &= (k^3 - k) + 3k^2 + 3k \\ &= 3q + 3(k^2 + k),\ q \in \mathbb{Z} \quad \text{(by the inductive hypothesis)} \\ &= 3(q + k^2 + k)\end{aligned}$$

So $3|((k+1)^3 - (k+1))$.

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

There is a clever direct proof without any induction for the above statement. Can you see it?

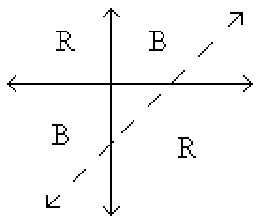
Two Color Theorem: There is a famous theorem called the four color theorem. It states that any map can be colored with four colors such that any two adjacent countries (which share a border, but not just a point) must have different colors. The four color theorem is very difficult to prove, and several bogus proofs were claimed since the problem was first posed in 1852. It was not until 1976 that the theorem was finally proved (with the aid of a computer) by Appel and Haken. (For an interesting history of the problem, and a state-of-the-art proof, which is nonetheless still very challenging, see `www.math.gatech.edu/ $\sim$thomas/FC/fourcolor.html`). We consider a simpler scenario, where we divide the plane into regions by drawing lines, where each line divides the plane into two regions (i.e. it extends to infinity). 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? Any border of a region either consists of a part of one of the original n lines or a piece of the $n+1$-st line. If it is a part of one of the original n lines, then the two regions on either side are both on the same side of the $n+1$-st line, and the colors of the regions must be distinct, by the induction hypothesis. On the other hand, if the border is part of the $n+1$-th line, then the two regions were created by dividing a single region from the induction hypothesis, and by construction we reversed colors on one side of the line, and so they have opposite colors. ♠

Induction is a very powerful technique. But you will need to exercise care while using it, since even small errors can lead to proving ridiculously false statements. Here is a dramatic example: in the middle of the last century, a colloquial expression in common use was "that is a horse of a different color", referring to something that is quite different from normal or common expectation. The famous mathematician George Polya (who was also a great expositor of mathematics for the lay public) gave the following proof to show that there is no horse of a different color!

Theorem: All horses are the same color.

Proof (by induction on the number of horses):

- Base Case: $P(1)$ is certainly true, since if you have a set containing just one horse, all horses in the set have the same color.
- Inductive Hypothesis: Assume $P(n)$, which is the statement that in any set of n horses, they all have the same color.
- Inductive Step: Given a set of $n+1$ horses $\{h_1, h_2, \ldots, h_{n+1}\}$, we can exclude the last horse in the set and apply the inductive hypothesis just to the first n horses $\{h_1, \ldots, h_n\}$, deducing that they all have the same color. Similarly, we can conclude that the last n horses $\{h_2, \ldots, h_{n+1}\}$ all have the same color. But now the "middle" horses $\{h_2, \ldots, h_n\}$ (i.e., all but the first and the last) belong to both of these sets, so they have the same color as horse h_1 and horse h_{n+1}. It follows, therefore, 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 induction proof? It is tempting to blame the induction hypothesis — which is clearly false. But the whole point of induction is that if the base case is true (which it is in this case), and assuming the induction hypothesis for any n we can prove the case $n+1$, then the statement is true for all n. So what we are looking for is a flaw in the reasoning!

What makes the flaw in this proof a little tricky to spot is that the induction step *is* valid for a "typical" value of n, say, $n=3$. The flaw, however, is in the induction step when $n=1$. In this case, for $n+1=2$ horses, there are *no* "middle" horses, and so the argument completely breaks down!

Strengthening the Inductive Hypothesis

Let us prove by induction the following proposition:

Theorem: $\forall n \geq 1$, the sum of the first n odd numbers is a perfect square.

Proof: By induction on n.

- Base Case: $n=1$. The first odd number is 1, which is a perfect square.
- Inductive Hypothesis: Assume that the sum of the first k odd numbers is a perfect square, say m^2.
- Inductive Step: The $k+1$-th odd number is $2k+1$, so by the induction hypothesis, the sum of the first $k+1$ odd numbers is m^2+2k+1. But now we are stuck. Why should m^2+2k+1 be a perfect square?

Well, let's just take a detour and compute the values of the first few cases. Maybe we will identify another pattern.

- $n=1: 1=1^2$ is a perfect square.
- $n=2: 1+3=4=2^2$ is a perfect square.
- $n=3: 1+3+5=9=3^2$ is a perfect square.
- $n=4: 1+3+5+7=16=4^2$ is a perfect square.

Wait, isn't there a pattern where the sum of the first n odd numbers is just n^2? Here is an idea: let us show something stronger!

Theorem: For all $n \geq 1$, the sum of the first n odd numbers is n^2.

Proof: By induction on n.

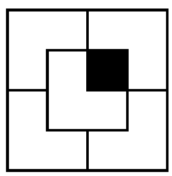
- Base Case: $n=1$. The first odd number is 1, which is 1^2.
- Inductive Hypothesis: Assume that the sum of the first k odd numbers is k^2.

- Inductive Step: The $(k+1)$-st odd number is $2k+1$, so by the induction hypothesis the sum of the first $k+1$ odd numbers is $k^2+(2k+1)=(k+1)^2$. Thus by the principle of induction the theorem holds. ♠

See if you can understand what happened here. We could not prove a proposition, so we proved a harder proposition instead! Can you see why that can sometimes be easier when you are doing a proof by induction? When you are trying to prove a stronger statement by induction, you have to show something harder in the induction step, but you also get to assume something stronger in the induction hypothesis. Sometimes the stronger assumption helps you reach just that much further...

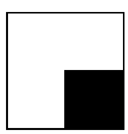
Here is another example:

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 tiling in the case that $n=2$:

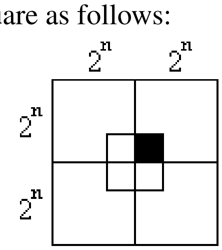


Let us try to prove the proposition by induction on n.

- 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:



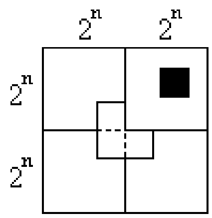
- Inductive Hypothesis: Assume $P(n)$ is true. That is, we can tile a $2^n\times 2^n$ courtyard with a missing 1×1 square in the middle.
- Inductive Step: We want to show that we can tile a $2^{n+1}\times 2^{n+1}$ courtyard with a missing 1×1 square in the middle. Let's try to reduce this problem so 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 inductive 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 we choose*. 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 just work on the inductive hypothesis and step.

- Inductive Hypothesis (second attempt): Assume $P(n)$ is true, so 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 the first tile as shown, we get four $2^n \times 2^n$ courtyards, each with a 1×1 hole; three of these courtyards have the hole in one corner, while the fourth has the hole in a position determined by the hole in the $2^{n+1} \times 2^{n+1}$ courtyard. The stronger inductive hypothesis now applies to each of these four courtyards, so that each of them 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! Hence, by the induction principle, we have proved the (stronger) theorem. ♠

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 $P(0)$, $P(1)$, $\ldots$, 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 $\bigwedge_{i=0}^{k} P(i)$ 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 $n \geq 2$ 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 $n = 2$. Clearly $P(2)$ 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 $x, y \in \mathbb{Z}^+$ and $1 < x, y < n+1$. By the inductive hypothesis, x and y can each 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 $P(41)$, i.e., that 41 can be written as a product of primes — a fact that is not useful in establishing $P(42)$.

To understand why strong induction works, let's think about our domino analogy. By the time we ready for the $k+1$-st domino to fall, dominoes numbered 0 through k have already been knocked over. But this is exactly what strong induction assumes: to prove $P(k+1)$, we can assume we already know that $P(0)$ through $P(k)$ are true.

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!). For the other direction, 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 $\forall k\, Q(k)$ by *simple* induction. The proof is modeled after the strong induction proof of $\forall n\, P(n)$. That is, we want to show $Q(k) \Rightarrow Q(k+1)$, or equivalently $P(0) \wedge \cdots \wedge P(k) \Rightarrow P(0) \wedge \cdots \wedge P(k) \wedge P(k+1)$. 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. And clearly, proving $\forall n\, Q(n)$ also proves $\forall n\, P(n)$.

Well Ordering Principle

In the context of proving statement about algorithms or programs, it is often convenient to formulate an induction proof in a different way. We start by asking how the statement $\forall n \in \mathbb{N}$, $P(n)$ could fail? Well, it means that there must be some values of n for which $P(n)$ is false. Let m be the smallest such natural number. We know that m must be greater than 0 since $P(0)$ is true (base case), which indicates $m-1 \in \mathbb{N}$. Since m is the smallest input that makes $P(m)$ false, $P(m-1)$ must be true. But $P(m-1) \rightarrow P(m)$, which is a contradiction.

We assumed something when defining m that is usually taken for granted: that we can actually find a smallest number in any subset of natural numbers. This property does *not* hold for, say, the real numbers; to see why, consider the set $\{x \in \mathbb{R} : 0 < x < 1\}$. For every number y in this set, we can always find a smaller one, e.g. $y/2$.

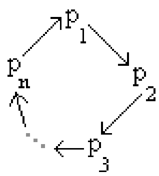
Formally, this property of natural numbers is called the well-ordering principle:

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

So the above argument by contradiction gives a valid alternative way of doing a proof by induction. In this form we start as before by proving the base case. But then we assume for contradiction that there is a smallest counterexample $P(k)$ (the well-ordering principle gives us a right to make this assumption). Now we show that this leads to a contradiction, thus completing the proof.

Let's look at an example.

Round robin tournament: Suppose that, in a round robin tournament, we have a set of k players $\{p_1, p_2, \ldots, p_k\}$ such that p_1 beats p_2, p_2 beats p_3, ..., p_{k-1} beats p_k, and p_k beats p_1. This is called a *cycle* in the tournament:

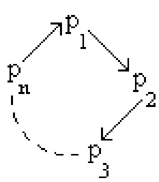


(A round robin tournament is a tournament where each participant plays every other contestant exactly once. Thus, if there are n players, there will be exactly $\frac{n(n-1)}{2}$ matches. Also, we are assuming that every match ends in either a win or a loss; no ties.)

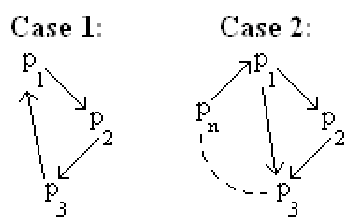
Claim: If there exists a cycle in a tournament, then there exists a cycle of length 3.

Proof: For the base case, notice that we cannot have a cycle of length less than 3, and if there is a cycle of length 3 then the proposition is true.

Assume for contradiction that the smallest cycle is:



with $n > 3$. Let us look at the game between p_1 and 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 $\{p_3, p_4, \ldots, p_n\}$ and thus a contradiction. Therefore, if there exists a cycle, then there must exist a 3-cycle as well. ♠



Can we prove this claim using more traditional induction? Let us start with the base case of $n = 3$ players and proceed from there.

Proof: By induction on n.

- Base Case: As above.
- Inductive Hypothesis: If a round-robin tournament has a cycle of length k then it has a cycle of length 3.

- Inductive Step: Given a round-robin tournament with a cycle of length $k+1$, we wish to show there must be a 3-cycle. Assume wlog that the cycle involves players p_1 through p_{k+1} in that order. Consider the outcome of the match between p_1 and p_3. If p_3 beats p_1 then we have a 3-cycle. If p_1 beats p_3, there is a k-cycle that goes directly from $p(1)$ $p(3)$ and continues as before. Applying the induction hypothesis, we conclude that there must be a 3-cycle in the tournament. ♠

Induction and Recursion

There is an intimate connection between induction and recursion in mathematics and computer science. 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: Starting with a pair of rabbits, how many rabbits do you end up with at the end of the year, if it is supposed that every month each pair begets a new pair which from the second month on becomes productive?

Let $F(n)$ denote the number of pairs of rabbits in month n. According to the above specification, the initial conditions are $F(0) = 0$ 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 $F(3) = 2$. 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 $n-1$ is $F(n-1)$. 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, in addition to the $F(n-1)$ already in existence. 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 $F(n) \geq 2^{(n-1)/2}$ for all $n \geq 3$?] 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 then return 0
  if n=1 then return 1
  else return F(n-1) + F(n-2)
```

Can you figure out how long this program takes 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 F2(n)
  if n=0 then return 0
  if n=1 then return 1
  a = 1
  b = 0
  for k = 2 to n do
    temp = a
    a = a + b
    b = temp
  return a
```

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

Clearly, induction and recursion are closely related. In fact, proofs involving a recursively-defined concept, e.g. factorial, are often best done using induction. Formally, the factorial of a nonnegative number n is defined recursively as $n! = n(n-1)(n-2)...1$, with a base case $0! = 1$, whereas exponentiation is defined recursively as $x^n = x^{n-1}x$. In this 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, we have chosen as our base case $n = 2$ rather than $n = 0$. This is because the statement is trivially true for $n < 2$.

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

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, $\forall n \in \mathbb{N}$, if $n > 1$, then $n! < n^n$. ♠

Practice Problems

1. Prove for any natural number n that $1^2 + 2^2 + 3^2 + \ldots + n^2 = \frac{1}{6}n(n+1)(2n+1)$.

2. Prove that $3^n > 2^n$ for all natural numbers $n \geq 1$.

3. In real analysis, Bernoulli's Inequality is an inequality which approximates the exponentiations of $1+x$. Prove this inequality, which states that $(1+x)^n \geq 1+nx$ if n is a natural number and $1+x > 0$.