Worksheet: Induction Proofs, II: Strong Induction. Application to Recurrences and Representation Problems

One of the most common applications of induction is to problems involving recurrence sequences such as the Fibonacci numbers, and to representation problems such as the representation of integers as a product of primes (Fundamental Theorem of Arithmetic), sums of powers of 2 (binary representation), sums of stamp denominations (postage stamp problem).

In applications of this type, the case $n = k$ in the induction step is not enough to deduce the case $n = k + 1$; one usually needs additional predecessors predecessors to get the induction step to work, e.g., the two preceding cases $n = k$ and $n = k - 1$, or *all* preceding cases $n = k, k - 1, \dots, 1$. This variation of the induction method is called **strong induction**. The induction principle remains valid in this modified form.

Strong Induction, I: Recurrences

For application of induction to two-term recurrence sequences like the Fibonacci numbers, one typically needs two preceding cases, $n = k$ and $n = k - 1$, in the induction step, and two base cases (e.g., $n = 1$ and $n = 2$) to get the induction going. The logical structure of such a proof is of the following form:

- **Base step:** $P(n)$ is true for $n = 1, 2$.
- **Induction step:** Let $k \in \mathbb{N}$ with $k \geq 2$ be given and assume $P(n)$ holds for $n = k$ and $n = k - 1$.
  [... Work goes here ...]
  Therefore $P(n)$ holds for $n = k + 1$.
- **Conclusion:** By the principle of strong induction, $P(n)$ holds for all $n \in \mathbb{N}$.

Remark: In the induction step, one could also say "Assume $P(n)$ holds for " $n = 1, 2, \dots, k$"; this is a bit redundant as only the last two of the cases $n = 1, 2, \dots, k$ are needed, though logically correct.

Sample strong induction proof: Recurrences

Claim: Let a_n be the sequence defined by $a_1 = 1$, $a_2 = 8$, and $a_n = a_{n-1} + 2a_{n-2}$ for $n \geq 3$. Prove that $a_n = 3 \cdot 2^{n-1} + 2(-1)^n$ for all $n \in \mathbb{N}$.
Proof: We will prove by strong induction that, for all $n \in \mathbb{N}$,

$$(*) \qquad a_n = 3 \cdot 2^{n-1} + 2(-1)^n.$$

Base case: When $n = 1$, the left side of $(*)$ is $a_1 = 1$, and the right side is $3 \cdot 2^0 + 2 \cdot (-1)^1 = 1$, so both sides are equal and $(*)$ is true for $n = 1$.
When $n = 2$, the left and right sides of $(*)$ are $a_2 = 8$ and $3 \cdot 2^1 + 2 \cdot (-1)^2 = 8$, so $(*)$ holds in this case as well.
Induction step: Let $k \in \mathbb{N}$ with $k \geq 2$ be given and suppose $(*)$ is true for $n = 1, 2, \dots, k$. Then

$$\begin{aligned} a_{k+1} &= a_k + 2a_{k-1} \quad \text{(by recurrence for } a_n) \\ &= 3 \cdot 2^{k-1} + 2 \cdot (-1)^k + 2\left(3 \cdot 2^{k-2} + 2 \cdot (-1)^{k-1}\right) \quad \text{(by } (*) \text{ for } n = k \text{ and } n = k-1) \\ &= 3 \cdot \left(2^{k-1} + 2^{k-1}\right) + 2\left((-1)^k + 2(-1)^{k-1}\right) \quad \text{(by algebra)} \\ &= 3 \cdot 2^k + 2(-1)^{k+1} \quad \text{(more algebra).} \end{aligned}$$

Thus, $(*)$ holds for $n = k + 1$, and the proof of the induction step is complete.
Conclusion: By the strong induction principle, it follows that $(*)$ is true for all $n \in \mathbb{N}$.

Strong Induction, II: Representation Problems

For applications to representation problems one typically requires the induction hypothesis in its strongest possible form, where one assumes *all* preceding cases (i.e., for $n = 1, 2, \dots, k$) instead of just the immediate predecessor.

Below is a classic example of this type, a proof that every integer ≥ 2 can be written as a product of prime numbers. This is the existence part of what is called the Fundamental Theorem of Arithmetic; the other part guarantees uniquess of the representation, which we will not be concerned with here.

The precise definitions of "prime" and "composite" are as follows: An integer $n \geq 2$ is called **composite** if it can be written as $n = ab$ with integers a, b satisfying $2 \leq a, b < n$. An integer $n \geq 2$ is called **prime** if it cannot be factored in this way. (Only integers ≥ 2 are classified in this way. In particular, the number 1 is neither prime nor composite.)

Sample strong induction proof: Fundamental Theorem of Arithmetic

Claim (Fundamental Theorem of Arithmetic, Existence Part): Any integer $n \geq 2$ is either a prime or can be represented as a product of (not necessarily distinct) primes, i.e., in the form $n = p_1 p_2 \dots p_r$, where the p_i are primes.
Proof: We will prove by strong induction that the following statement holds for all integers $n \geq 2$.

$(P(n))$ n can be represented as a product of one or more primes.

Base case: The integer $n = 2$ is a prime since it cannot be written as a product ab, with integers $a, b \geq 2$, so $P(n)$ holds for $n = 2$.
Induction step:

- Let $k \geq 2$ be given and suppose $P(n)$ is true for all integers $2 \leq n \leq k$, i.e., suppose that all such n can be represented as a product of one or more primes.
- We seek to show that $k + 1$ also has a representation of this form.
- If $k + 1$ itself is prime, then $P(n)$ holds for $n = k + 1$, and we are done.
- Now consider the case when $k + 1$ is composite.
- By definition, this means that $k + 1$ can be written in the form $k + 1 = ab$, where a and b are integers satisfying $2 \leq a, b < k + 1$, i.e., $2 \leq a, b \leq k$.
- Since $2 \leq a, b \leq k$, the induction hypothesis can be applied to a and b and shows that a and b can be represented as products of one or more primes.
- Multiplying these two representations gives a representation of $k + 1$ as a product of primes.
- Hence $k + 1$ has a representation of the desired form, so $P(n)$ holds for $n = k + 1$, and the induction step is complete.

Conclusion: By the strong induction principle, it follows that $P(n)$ is true for all $n \geq 2$, i.e., every integer $n \geq 2$ is either a prime or can be represented as a product of primes.

Advice on writing up induction proofs

- **Strong versus ordinary induction: Which method should you use?** With some standard types of problems (e.g., sum formulas) it is clear ahead of time what type of induction is *likely* to be required, but usually this question answers itself during the exploratory/scratch phase of the argument. In the induction step you will need to reach the $k+1$ case, and you should ask yourself which of the previous cases you need to get there. If all you need to prove the $k+1$ case is the case k of the statement, then ordinary induction is appropriate. If two preceding cases, $k-1$ and k, are necessary to get to $k+1$, then (a weak form of) strong induction is appropriate. If one needs the full range of preceding cases (i.e., all cases $n = 1, 2, \dots, k$), then the full force of strong induction is needed.

- **Base step: How many base cases are needed?** The number of base cases to be checked depends on how far back one needs to "look" in the induction step. In standard induction proofs (e.g., for summation formulas) the induction step requires only the immediately preceding case (i.e., the case $n = k$), so a single base case is enough to start the induction.

  - For Fibonacci type problems, the induction step usually requires the result for the two preceding cases, $n = k$ and $n = k - 1$. To get the induction started, one therefore needs to know the result for two consecutive cases, e.g., $n = 1$ and $n = 2$.
  - In postage stamp type problems, getting the result for $n = k + 1$ might require knowing the result for $n = k - 2$ and $n = k - 6$, say. This amounts to "looking back" 7 steps (namely $n = k, k - 1, \dots, k - 6$), so 7 consecutive cases are needed to get the induction started.
  - On the other hand, in problems involving the full strength of the strong induction hypothesis (i.e., if in the induction step one needs to assume the result for *all* preceding cases $n = k, k - 1, \dots, 1$), a single base case may be sufficient. An example is the Fundamental Theorem of Arithmetic.

- **Write-up of the induction step.** As in the case of ordinary induction, at the beginning of the induction step **state precisely what you are assuming, including any constraints on the induction variable** k. Without an explicitly stated assumption, the argument is incomplete. The appropriate induction hypothesis depends on the nature of the problem and the type of induction used. Here are some common ways to start out an induction step:

  - "Let $k \in \mathbb{N}$ be given and assume $(*)$ is true for $n = k$." (typical form for standard induction proofs)
  - "Let $k \geq 2$ be given and assume $(*)$ holds for $n = k - 1$ and $n = k$." (typical form for induction involving recurrences)
  - "Let $k \in \mathbb{N}$ be given and assume $P(n)$ holds for $n = 1, 2, \dots, k$." (typical form for representation problems)

Practice Problems

1. **Recurrences:** The first few problems deal with properties of the Fibonacci sequence and related recurrence sequences. The Fibonacci sequence is defined by $F_1 = 1$, $F_2 = 1$, and $F_n = F_{n-1} + F_{n-2}$ for $n \geq 3$. Its first few terms are $1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, \ldots$.

   In the following problems, use an **appropriate** form of induction (standard induction or strong induction) to establish the desired properties and formulas. (Note that some of these problems require only ordinary induction.)

   (a) **Fibonacci sums:** Prove that $\sum_{i=1}^{n} F_i = F_{n+2} - 1$ for all $n \in \mathbb{N}$.

   (b) **Fibonacci matrix:** Let A be the 2×2 matrix $A = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 0 \end{smallmatrix}\right)$. Let $A^n = \left(\begin{smallmatrix} a_n & b_n \\ c_n & d_n \end{smallmatrix}\right)$ denote this matrix multiplied by itself n times. Find and prove a formula for the four entries a_n, b_n, c_n, d_n.

   (c) **Odd/even Fibonacci numbers:** Prove that the Fibonacci numbers follow the pattern odd,odd,even: that is, show that for any positive integer m, F_{3m-2} and F_{3m-1} are odd and F_{3m} is even.

   (d) **Inequalities for recurrence sequencs:** Let the sequence T_n ("Tribonacci sequence") be defined by $T_1 = T_2 = T_3 = 1$ and $T_n = T_{n-1} + T_{n-2} + T_{n-3}$ for $n \geq 4$. Prove that

   $$(*) \qquad T_n < 2^n$$

   holds for all $n \in \mathbb{N}$.

2. **Representation problems.** One of the main applications of strong induction is to prove the existence of representations of integers of various types. In these applications, strong induction is usually needed in its full force, i.e., in the induction step, one needs to assume that all predecessor cases $n = 1, 2, \ldots, k$.

   (a) **The postage stamp problem:** Determine which postage amounts can be created using the stamps of 3 and 7 cents. In other words, determine the exact set of positive integers n that can be written in the form $n = 3x + 7y$ with x and y nonnegative integers. (Hint: Check the first few values of n directly, then use strong induction to show that, from a certain point n_0 onwards, all numbers n have such a representation.)

   (b) **Binary representation:** Using strong induction prove that every positive integer n can be represented as a sum of *distinct* powers of 2, i.e., in the form $n = 2^{i_1} + \cdots + 2^{i_h}$ with integers $0 \leq i_1 < \cdots < i_h$. (Hint: To ensure distinctness, use the *largest* power of 2 as the first "building block" in the induction step.)