Recursively Defined Functions and Sets, Structural Induction

Recursively Defined Functions

A recursively defined function is a function whose definition refers back to itself. A classic example of such a function is the factorial,

$$n! = 1 \cdot 2 \cdot 3 \cdot \cdots \cdot n,$$

which can be defined recursively as follows:

$$0! = 1$$
$$n! = n(n-1)! \text{ for all } n \in \mathbb{N}.$$

The Ackermann function

A much more difficult example of a recursively defined function is the Ackermann function, which is defined for all $n, m \in \mathbb{N}_0$:

$$A(m,n) = \begin{cases} n+1 & \text{if } m = 0 \\ A(m-1,1) & \text{if } m > 0 \text{ and } n = 0 \\ A\big(m-1, A(m,n-1)\big) & \text{if } m > 0 \text{ and } n > 0 \end{cases}$$

Observe that evaluation of $A(m,n)$ is direct (not involving self-reference) only when $m = 0$. For higher values of m, the definition requires us to find other function values first. Example:

$$A(1,1) = \mathrm{A}(0, \mathrm{A}(1,0))$$
$$A(1,0) = A(0,1) = 2$$

Therefore

$$A(1,1) = A(0,2) = 3.$$

Recursively Defined Sets (I)

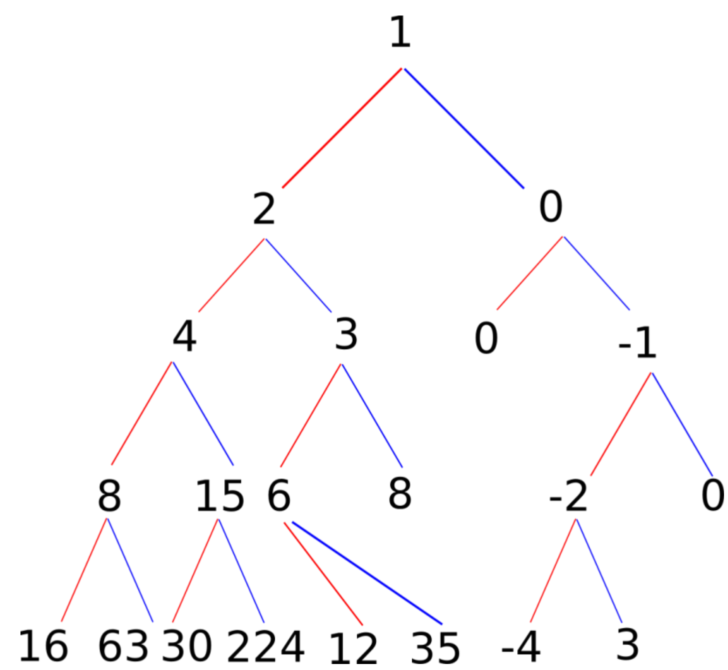
We will study an example of a recursively defined set before we give a general definition. Let S be defined by the following properties:

1. $1 \in S$
2. if $x \in S$, then $2x \in S$
3. if $x \in S$, then $(x^2 - 1) \in S$

Since $1 \in S$, condition 2 implies $2 \in S$, and condition 3 implies $0 \in S$. A tree diagram (on the right) is helpful to explore the membership of S more systematically.

As we can see, S contains the numbers $\pm 2^n$ for all $n \in \mathbb{N}_0$, but also other “random” integers. There is no obvious direct (non-recursive) condition that describes this set.

Observe that in a recursively defined set, there is no natural linear order- no “first”, “second”, etc. element.



red: application of rule 2
blue: application of rule 3

Recursively Defined Sets (II)

For the recursively defined set S from the previous slide, let us prove that $5 \notin S$.

Proof by contradiction: Suppose $5 \in S$. Since $5 \neq 1$, $5 \in S$ by property 2 or 3. Therefore, $5 = 2x$ for some $x \in S$, or $5 = x^2 - 1$ for some $x \in S$. Since S consists only of integers, $5 = 2x$ is not possible for any $x \in S$. Therefore, $5 = x^2 - 1$ for some $x \in S$, which implies $x^2 = 6$. Again, since S only contains integers, that is not possible. This is a contradiction, and it completes the proof.

Observe that our proof relies on an "obvious" property of S: all elements of S are integers. To prove that, we need a proof technique that allows us to prove statements that are true for all elements in a recursively defined set. That technique is **structural induction**.

A formal description of Recursively Defined Sets and Structural Induction

A **recursively defined set** is a set S that is defined as follows:

1. All members of a given set G are in S.
2. For some given functions f_k, whenever $s \in S$, then $f_k(s) \in S$ for all k.

Structural Induction is a way of proving that all elements of a recursively defined set have a certain property. It is based on the following **principle of structural induction**:

Suppose a set S is recursively defined as above. Suppose $P(x)$ is a statement for all $x \in S$. If $P(x)$ is true for all $x \in G$, and furthermore, if $P(x) \rightarrow P(f_k(x))$ is true for all $x \in S$ and all f_k, then $P(x)$ is true for all $x \in S$.

We could think of a recursively defined set as a population, and the set G as the initial *population*. The functions f_k could be thought of as rules of procreation. Then, the principle of structural induction says that if something is true about everyone in the initial population, and if every individual always passes it on to their direct descendants, then it is true about the entire population.

Examples of Recursive Descriptions of Common Sets (1)

$\mathbb{N} = \{1,2,3, \ldots\}$, the set of natural numbers, can be recursively defined as follows:

$$1 \in \mathbb{N}$$
$$n \in \mathbb{N} \rightarrow n + 1 \in \mathbb{N}.$$

The set of integers $\mathbb{Z}$ can be recursively defined by:

$$0 \in \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow n - 1 \in \mathbb{Z}.$$

There are many other ways of defining $\mathbb{Z}$ recursively. Here is another:

$$0 \in \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow n + 1 \in \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow -n \in \mathbb{Z}$$

And yet another:

$$\mathbb{N} \subseteq \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow -101n \in \mathbb{Z}$$
$$n \in \mathbb{Z} \rightarrow 29n \in \mathbb{Z}$$
$$n, m \in \mathbb{Z} \rightarrow n + m \in \mathbb{Z}$$

Understanding why the last one defines $\mathbb{Z}$ is intended to be a challenge.

Examples of Recursive Descriptions of Common Sets (2)

Here is a possible recursive definition for the set of rational numbers $\mathbb{Q}$:

$$\mathbb{Z} \subseteq \mathbb{Q}$$
$$n, m \in \mathbb{Q} \land m \neq 0 \rightarrow \frac{n}{m} \in \mathbb{Q}.$$

Here is another one:

$$0 \in \mathbb{Q}$$
$$n \in \mathbb{Q} \rightarrow n + 1 \in \mathbb{Q}$$
$$n \in \mathbb{Q} \rightarrow n - 1 \in \mathbb{Q}$$
$$n, m \in \mathbb{Q} \land m \neq 0 \rightarrow \frac{n}{m} \in \mathbb{Q}.$$

Here is yet another, less obvious one:

$$1 \in \mathbb{Q}$$
$$n, m \in \mathbb{Q} \land m \neq 0 \rightarrow \frac{n}{m} \in \mathbb{Q}$$
$$n, m \in \mathbb{Q} \rightarrow n - m \in \mathbb{Q}.$$

Structural Induction Described Using a Population Metaphor

If the initial population all had a trait, and if that trait is always passed down to all direct descendants, then the entire population has that trait.

Therefore, to prove that an entire population has a trait, you need to verify only that

1. The initial population had the trait.
2. The trait is always transmitted to all direct descendants.

Structural Induction Proof 1

When we previously proved that 5 was not an element of the set S, we used the "obvious" fact that all elements of S are integers. We will now prove this using structural induction.

Recall the definition of S:

1. $1 \in S$
2. if $x \in S$, then $2x \in S$
3. if $x \in S$, then $(x^2 - 1) \in S$

Theorem: $S \subseteq \mathbb{Z}$ (i.e., all elements of S are integers.)

Proof by structural induction: every member of the "initial population" $\{1\}$ is an integer. Now suppose that $x \in S$ is an integer. Then $2x$ and $x^2 - 1$ are integers as well. Therefore, by the principle of structural induction, all elements of S are integers.

(We proved that the initial population only consisted of integers, and that the property of being integer was passed down through all the rules that create "new" elements from "old" elements. Therefore, the entire population consists of integers.)

Structural Induction Proof 2

Let S be recursively defined as follows:

1. $2 \in S$
2. if $n \in S$, then $3n \in S$
3. if $n \in S$, then $n^2 \in S$.

Prove that all members of S are even.

Base case: the initial population, 2, is even.

Inductive Step: suppose some arbitrary $n \in S$ is even. By the rules for even and odd numbers, then $3n$ and n^2 must also be even.

By the principle of structural induction, all elements of S are therefore even.

Structural Induction Proof 3

Suppose a population of monkeys was set free on an island, and all monkeys carried a virus. Biologists have shown that when a monkey carries the virus, all its offspring will permanently carry it too.

Proof by Structural Induction that all monkeys on the island carry the virus:

Base Case: The initial population of monkeys on the island carried the virus.

Inductive step: now suppose you have an infected monkey. Since the infection is 100% contagious, all of its direct offspring also have the infection.

By the principle of structural induction, all monkeys on the island therefore carry the virus.

Common Mistakes in Structural Induction Proofs (I)

Don't confuse structural induction with regular (linear) induction.

A recursively defined set is in general not linear, i.e. there is no single "next" element for each element. It’s a “tree”, not a “line”. Elements in a recursively defined set generally have multiple "next" elements.

Even the concept of “next” elements (plural) is questionable. The same element can occur repeatedly in the tree, as you can see in the example on page 4. In that example, 3 occurs in the second generation, but also in the 5th generation.

This means that any talk of $P(n)$ implying $P(n+1)$ has no place in a structural induction proof. There is no n-th statement and no n-th element because there is no linear order to the set. The elements of the recursively defined set are not naturally indexed by the positive integers, and no such indexing may even exist.

But can't we force a linear structure on a recursively defined set by looking at the elements that have been created by n applications of the rules, for every n?

This sounds like a reasonable idea. Let's say S_n is the subset of the recursively defined set S that can be created by applying the recursion rules of the set to the initial population up to n times.

S_0 is then the initial population. In the inductive step, we would then assume that we already have proved our statement for all members of S_n, and must then show that it also holds for all members of S_{n+1}.

Think about what carrying out this plan means in practice. Verifying the statement for S_0 is verifying it for the initial population, which is exactly what we do in a structural induction proof.

Performing the inductive step requires us to show that if a member x of S_n has the property, then so do its descendants. All we know about x is that it is in S and has the property in question. From that, we show that the descendants have the property too. <u>At no point in that argument will the quantity n have any relevance.</u> It doesn't matter whether *x* is the product of two, or 999 applications of the recursion rules. The only thing that matters is that x has the property we are proving all members of S have.

This means that if you use this approach, you end up doing structural induction anyway, just with the added distraction and confusion of a wholly redundant framework of subsets S_n and an irrelevant parameter n that will tempt you to force it somehow – and guaranteed incorrectly - into your algebraic argument.

Common Mistakes in Structural Induction Proofs (II)

There is another misunderstanding by which students sometimes try to shoehorn a recursively defined set into a linear structure: assuming that the operations which create new elements from old ones have to be applied in the sequence in which they were listed in the definition, and then repeated.

Referring back to structural induction proof 2, they might think that the two operations of multiplication by 3 and squaring are to be applied alternatingly, starting with the initial population, and that the set is therefore the sequence 2, 6, 36, 108, 11664, 34992, …. .

This can lead to the belief (which is illogical even if you make the assumption we just described) that in the inductive step, you have to verify only that the composition of all the rules preserves the property in question. We illustrate this with the example of structural induction proof 2:

(Incorrect) Inductive Step: suppose some arbitrary $n \in S$ is even. Then $(3n)^2$ must also be even.

This is both insufficient and redundant. It's insufficient because showing that two rules in combination preserve the property in question does not imply that the two rules do so individually; and it's redundant because once you have shown that the two rules preserve the property in question individually, you automatically know that they preserve it in combination.

Common Mistakes in Structural Induction Proofs (III)

The base case consists of verifying **only** that the initial population has the desired property, **not** the first-generation descendants.

Not only does a discussion of the first-generation descendants not belong in the base case, it also does not – specifically as such - belong in the inductive step. <u>The inductive step, properly executed, takes care of all descendants simultaneously.</u>

The following is an <u>incorrect</u> execution of the base case in structural induction proof 2:

"*Base case: The initial population, 2, is even. When we apply rule 2 to that, we get 6, and when we apply rule 3, we get 4. Both numbers are even.* "

The part marked red is technically not wrong, it's just redundant and therefore not properly part of the base case.

Common Mistakes in Structural Induction Proofs (V)

Do not substitute an intuitive global explanation for the strictly local perspective of a structural induction proof. Compare example 3 with the following incorrect version:

Not a structural induction argument: *The initial population of monkeys on an island carried a virus. Offspring of infected monkeys will always carry the virus as well. Therefore, the second generation of monkeys also carry the virus, and therefore, the third generation too, and so on and so forth. Thus, all monkeys on the island carry the virus.*

Notice that this argument is not rigorous. It appeals to the intuition of the reader at the "and so on" stage. The reader is asked to believe that the pattern continues. It is precisely this aspect which structural induction makes rigorous. The principle of structural induction, once generally proved, guarantees that whenever an initial population has a property, and that property is 100% contagious, all descendants are going to have that property. A proof that builds on this principle is unconcerned with re-proving or re-explaining the global logic of this principle. It only verifies that its assumptions are true, and that verification is strictly local: it only verifies that the initial population has the property, and that whenever one <u>arbitrary</u> member has the property, its <u>direct</u> offspring has the property too.

Do not try to re-prove or re-explain the global logic of structural induction in a structural induction proof. If at any point you are referring to the direct descendants of the initial population, or referring to the initial population in the inductive step, or making a "and the pattern continues" type argument, then you're not writing a structural induction proof.

Common Mistakes in Structural Induction Proofs (VI)

Another common mistake is tying the inductive step to the base case. The following incorrect version of the inductive step of structural induction proof 2 illustrates this mistake.

Inductive Step: suppose some arbitrary $n \in S$ is even. We must then show that $3n$ and n^2 must also be even. By the base case, $n = 2$, so $3n = 6$ and $n^2 = 4$. These are both even. That completes the proof by structural induction.

Induction is about setting up a logical chain reaction that propagates through the entire set. Each conclusion serves as the hypothesis trigger for the next conclusion(s). This cannot happen if you artificially limit the inductive hypothesis to the base case. The chain reaction then dies after one step.

The inductive step needs to work for an **arbitrary** element of the recursively defined set. It must not be tied to a specific element. ANY reference to the base case in the inductive step is an indication that there is something wrong with your logic.

Common Mistakes in Structural Induction Proofs (VII)

Another common mistake is confusing the definition of the recursively defined set S with what we are trying to prove *about* the set. Here is an <u>incorrect</u> fragment of the inductive step of structural induction proof 2 that illustrates this mistake.

Inductive Step: suppose some arbitrary $n \in S$ *is even. (..) We have shown that* $3n$ *and* n^2 *are also in* S*.*

That those two elements are in S is part of the definition of S and requires no proof.

Common Mistakes in Structural Induction Proofs (VIII)

A final common mistake is assuming the conclusion in the inductive hypothesis.

Inductive Step: suppose we have some some arbitrary $n \in S$. *Then* n *is even.*

This inductive step assumes the conclusion. If we already know that an arbitrary $n \in S$ is even, then there is nothing to prove.

That the given $n \in S$ of the inductive step is even is part of the **assumption** of the inductive step.

A More Substantial Proof Involving Structural Induction

Let S be recursively defined as follows:
1. $1 \in S$
2. if $x \in S$, then $5x \in S$
3. if $x \in S$, then $3x \in S$

Based on the definition, we see that 3 and 5 are in S, therefore $3^2, 3 \cdot 5$ and 5^2 as well, etc. We guess that the following must be true:

Theorem: $S = \{3^i 5^j | i, j \in \mathbb{N}_0\}$. Proof: we have to show two statements,

I. $S \subseteq \left\{3^i 5^j \middle| i, j \in \mathbb{N}_0\right\}$: Here we have to show that every element in S is of the form $3^i 5^j$ with $i, j \in \mathbb{N}_0$. We will show this by structural induction. Certainly, each element of the initial population has that form: $1 = 3^0 5^0$.

Now suppose that $x \in S$ is equal to $3^i 5^j$ for some $i, j \in \mathbb{N}_0$. Then $3x = 3^{i+1} 5^j$ and $5x = 3^i 5^{j+1}$. These numbers again have the form a non-negative power of 3 times a non-negative power of 5. Therefore, by the principle of structural induction, all elements of S have that form.

II. $S \supseteq \left\{3^i 5^j \middle| i, j \in \mathbb{N}_0\right\}$: Now we have to show that every number of the form $3^i 5^j$ with $i, j \in \mathbb{N}_0$ is in S. We show this **constructively**, i.e. by stating how we get a number $3^i 5^j$ by applying the rules 1., 2. and 3. Given $n = 3^i 5^j$ with $i, j \in \mathbb{N}_0$,

$$n = 1 \cdot \underbrace{3 \cdot \dots \cdot 3}_{i \text{ times}} \cdot \underbrace{5 \cdot \dots \cdot 5}_{j \text{ times}}$$

This shows that by starting with the number 1 which we know is in S and applying rule 2. i times, then applying rule 3. j times, we get that $n \in S$.

A Structural Induction Proof Involving Bit Strings

Let a set S of bit strings be recursively defined as follows:

1. $00 \in S$
2. if $b \in S$, then $11b \in S$
3. if $b \in S$, then $0b0 \in S$.

(Note that bit strings are *strings* and that strings don't have a multiplication defined on them. If a and b are strings, then ab is the *concatenation* of a and b.)

Theorem: all elements of S end in at least two zeros.

Base case: 00 ends in two zeros.

Inductive Step: Now suppose that $b \in S$ ends in at least two zeros, i.e. $b = c00$ for some bit string c. Then:

- $11b = 11c00$ ends in at least two zeros, and
- $0b0 = 0c000$ ends in at least three zeros, therefore also in at least two zeros.

This completes the proof by structural induction.

Structural Induction for Recursively Defined Sequences (1)

We can use the principle of structural induction to prove statements about recursively defined sequences. This may be surprising, because structural induction is for proving statements about recursively defined *sets*. Recall that a set is an unordered data structure, whereas a sequence is an infinite list, which is an ordered data structure. If you reduce a sequence to its set of values (its *range*), you lose the sequence character.

For example, if you only know that the range of a sequence is {0,1}, that does not give you enough information to reconstruct the sequence. Indeed, there are infinitely many sequences that have that range. Here are two of them:

$a_n = (-1)^n$ for $n \geq 0$ (the sequence 1, 0, 1, 0, 1, …)

$b_n = \begin{cases} 0 \text{ for } n = 0 \\ 1 \text{ for } n \geq 1 \end{cases}$ (the sequence 0, 1, 1, 1, 1, …)

It stands to reason then that if you reduce a sequence $\{a_n\}$ to its range, you can at best prove statements about the range, but you cannot possibly derive or prove a closed form for a_n, since different sequences may have the same range.

How do we then encode a sequence as a set without losing the ordering information?

Structural Induction for Recursively Defined Sequences (2)

The answer to the question on the previous page lies in using ordered pairs of index values and corresponding sequence terms. So we think of a sequence $\{a_n\}_{n\in I}$ as the set $S = \{(n, a_n): n \in I\}$, where I is the index set of the sequence, usually $\mathbb{N}$ or $\mathbb{N}_0$.
For example, the sequence defined by $a_n = n^2$ for $n \geq 0$ then becomes the set $\{n^2 \mid n \geq 0\} = \{(0,0), (1,1), (2,4), (3,9), \dots.\}$. If you think back to college algebra, this is actually a familiar notion – it is the set of points "(x,y)" of the function, also known as the *graph*.
For a recursively defined sequence, initial value and recursion rules can be rewritten in this set language to produce a recursively defined set. Keep in mind that a sequence term a_n is now (n, a_n). Let us illustrate this with an example. Let $a_0 = 1$ and $a_{n+1} = 2a_n$ for all $n \geq 0$. Then the graph is $S = \{(0,1), (1,2), (2,4), (3,8), (4,16), \dots\}$. The recursive definition we derive directly from the set definition is:

$$(0,1) \in S$$
$$(n, x) \in S \rightarrow (n + 1, 2x) \in S.$$

Structural Induction for Recursively Defined Sequences (3)

Let us show two multi-step examples:

Example 1: $a_0 = 1$, $a_1 = 4$ and $a_{n+2} = 3a_{n+1} - 2a_n$ for all $n \geq 0$. The corresponding recursive set description is

$$(0,1) \in S$$
$$(1,4) \in S$$
$$(n, x) \in S \wedge (n+1, y) \in S \to (n+2, -2x + 3y) \in S.$$

Example 2: $a_0 = 0$, $a_1 = 2$ and $a_{n+2} = na_{n+1} - n^2 a_n$ for all $n \geq 0$. The corresponding recursive set description is

$$(0,0) \in S$$
$$(1,2) \in S$$
$$(n, x) \in S \wedge (n+1, y) \in S \to (n+2, -n^2 x + ny) \in S.$$

Structural Induction for Recursively Defined Sequences (4)

Now let us use structural induction to prove statements about recursively defined sets. We will use our first example, the geometric sequence $\{a_n\}$ whose graph S is recursively defined by

$$(0,1) \in S$$
$$(n, x) \in S \rightarrow (n+1, 2x) \in S.$$

We will prove $a_n = 2^n$ for all $n \geq 0$ by structural induction. This means we have to verify $x = 2^n$ for all $(n, x) \in S$.

Base case: the initial population$(n, x) = (0,1)$ satisfies $x = 2^n$ because $1 = 2^0$.

Inductive step: suppose we already know that some arbitrary $(n, x) \in S$ satisfies $x = 2^n$. Now we wish to show that its sole direct descendant, $(n+1, 2x)$, satisfies it too. Using our inductive hypothesis $x = 2^n$, we get $2x = 2 \cdot 2^n = 2^{n+1}$. That is the equation we needed to verify about $(n+1, 2x)$.

Structural Induction for Recursively Defined Sequences (5)

Let us prove a closed form for an arithmetic sequence. Let $\{a_n\}$ be defined by $a_0 = 3$ and $a_{n+1} = a_n + 5$ for all $n \geq 0$. The graph S is recursively defined by

$$(0,3) \in S$$
$$(n, x) \in S \rightarrow (n + 1, x + 5) \in S.$$

We will prove $a_n = 5n + 3$ for all $n \geq 0$ by structural induction. This means we have to verify $x = 5n + 3$ for all $(n, x) \in S$.

Base case: the initial population $(n, x) = (0,3)$ satisfies $x = 5n + 3$ because $3 = 5 \cdot 0 + 3$.

Inductive step: suppose we already know that some arbitrary $(n, x) \in S$ satisfies $x = 5n + 3$. Now we wish to show that its sole direct descendant, $(n + 1, x + 5)$, satisfies the condition. Using our inductive hypothesis $x = 5n + 3$, we get $x + 5 = 5n + 3 + 5 = 5(n + 1) + 3$. That is the equation we needed to verify about $(n + 1, x + 5)$.

Structural Induction for Recursively Defined Sequences (6)

Let us now prove a closed form for the two-step recursion we have previously introduced: $a_0 = 1$, $a_1 = 4$ and $a_{n+2} = 3a_{n+1} - 2a_n$ for all $n \geq 0$. Recall the corresponding recursive set description as

$$(0,1) \in S$$
$$(1,4) \in S$$
$$(n,x) \in S \land (n+1,y) \in S \rightarrow (n+2,-2x+3y) \in S.$$

We will prove $a_n = 3 \cdot 2^n - 2$ for all $n \geq 0$ by structural induction. This means we have to verify the condition $x = 3 \cdot 2^n - 2$ for all $(n,x) \in S$ by structural induction. Base case:

$(n,x) = (0,1)$ satisfies $x = 3 \cdot 2^n - 2$ because $1 = 3 \cdot 2^0 - 2$.
$(n,x) = (1,4)$ satisfies $x = 3 \cdot 2^n - 2$ because $4 = 3 \cdot 2^1 - 2$.

Inductive step: suppose we already know that some arbitrary $(n,x) \in S$ and $(n+1,y) \in S$ satisfy the condition. This means that $x = 3 \cdot 2^n - 2$ and $y = 3 \cdot 2^{n+1} - 2$.

Now we wish to show that $(n+2,-2x+3y)$ satisfies the condition. We need to verify that $-2x + 3y = 2^{n+2}$. We do this by starting with the left side and substituting the hypotheses about x and y:

$$-2x + 3y = -2(3 \cdot 2^n - 2) + 3(3 \cdot 2^{n+1} - 2) = -3 \cdot 2^{n+1} + 4 + 9 \cdot 2^{n+1} - 6$$

This simplifies to $-2x + 3y = 6 \cdot 2^{n+1} - 2 = 3 \cdot 2 \cdot 2^{n+1} - 2 = 3 \cdot 2^{n+2} - 2$.

Structural Induction for Recursively Defined Sequences (7)

Observe that behind the formality of the structural induction argument on the graph of recursively defined sequence with step size k, the inductive argument boiled down to **showing the desired condition for the initial case(s), and then showing that whenever the condition holds for k cases in a row, then it also holds for the next case.** For $k = 1$, this is just the principle of mathematical induction as previously introduced. For general k, it is a generalization of this principle. Let us call it the *generalized principle of induction*.

Let us simplify our previous proof by referring to this generalized principle of induction we have now recognized.

Base case:

$a_n = 3 \cdot 2^n - 2$ is satisfied for $n = 0$ and $n = 1$ because $1 = 3 \cdot 2^0 - 2$ and $4 = 3 \cdot 2^1 - 2$.

Inductive step:

Assume $a_n = 3 \cdot 2^n - 2$ and $a_{n+1} = 3 \cdot 2^{n+1} - 2$ for some arbitrary $n \geq 0$.

Then $a_{n+2} = 3a_{n+1} - 2a_n$ by definition of the sequence. Substituting our hypotheses and simplifying yields

$$a_{n+2} = 3a_{n+1} - 2a_n = 3(3 \cdot 2^{n+1} - 2) - 2(3 \cdot 2^n - 2) = -3 \cdot 2^{n+1} + 4 + 9 \cdot 2^{n+1} - 6.$$

Further simplification produces

$a_{n+2} = -3 \cdot 2^{n+1} + 4 + 9 \cdot 2^{n+1} - 6 = 6 \cdot 2^{n+1} - 2 = 3 \cdot 2 \cdot 2^{n+1} - 2 = 3 \cdot 2^{n+2} - 2$. This completes the inductive step.

We have thus proved $a_n = 3 \cdot 2^n - 2$ for all $n \geq 0$ by the generalized principle of induction.

Application: Proving An Estimate for the Fibonacci Sequence

Let us recall that the Fibonacci Sequence is the sequence $\{f_n\}$ defined by the initial conditions $f_0 = 1$, $f_1 = 1$ and the recurrence relation $f_n = f_{n-1} + f_{n-2}$ for all integers $n \geq 2$.

Let us prove that $f_n \geq \left(\frac{3}{2}\right)^{n-1}$ for all $n \geq 0$.

This is an interesting statement because it demonstrates that the Fibonacci sequence grows at least exponentially.

We will prove this by generalized induction.

Fibonacci Sequence, continued

First we need to verify $f_n \geq \left(\frac{3}{2}\right)^{n-1}$ for $n = 0$ and $n = 1$:

$$f_0 = 1 \geq \left(\frac{3}{2}\right)^{-1} = \frac{2}{3}; \qquad f_1 = 1 \geq \left(\frac{3}{2}\right)^{0} = 1$$

Both of these inequalities hold. Now that the base case is concluded, we turn our attention to the inductive step. Let us assume that the inequality has already been verified for some arbitrary n and its successor $n + 1$. We now show that it holds for $n + 2$. By definition of the sequence and by the assumption,

$$f_{n+2} = f_{n+1} + f_n \geq \left(\frac{3}{2}\right)^{n} + \left(\frac{3}{2}\right)^{n-1}.$$

We now rewrite the expression on the right as follows:

$$\left(\frac{3}{2}\right)^{n} + \left(\frac{3}{2}\right)^{n-1} = \left(\frac{3}{2}\right)^{n}\left(1 + \frac{2}{3}\right) = \left(\frac{3}{2}\right)^{n} \cdot \frac{5}{3} \geq \left(\frac{3}{2}\right)^{n} \cdot \frac{3}{2} = \left(\frac{3}{2}\right)^{n+1}.$$

Therefore, we have shown $f_{n+2} \geq \left(\frac{3}{2}\right)^{n+1}$, which was the desired statement.

An Improved Estimate for the Fibonacci Sequence (1)

We can go a little higher for the base in our estimate than $\frac{3}{2}$. Observe that in order to make the last inequality work, we used $\frac{5}{3} > \frac{3}{2}$. That suggests that there is some wiggle room left to increase the $\frac{3}{2}$. We could experiment with fractions higher than $\frac{3}{2}$, but let us take a systematic approach instead, and try to prove a generalization of our inequality, i.e. $f_n \geq a^{n-1}$ for all $n \geq 0$, where $a \geq \frac{3}{2}$ is some number yet to be determined.

The inequality holds for $n = 0$ and $n = 1$, because $a^0 = 1 = f_0$ and $a^{-1} \leq \frac{2}{3} < 1 = f_1$. In the inductive step, we need the generalization of

$$\left(\frac{3}{2}\right)^n + \left(\frac{3}{2}\right)^{n-1} \geq \left(\frac{3}{2}\right)^{n+1}$$

to hold, which is $a^n + a^{n-1} \geq a^{n+1}$. Since a is positive, that inequality is equivalent to $a^1 + a^0 \geq a^2$, or $a^2 - a - 1 \leq 0$.

An Improved Estimate for the Fibonacci Sequence (2)

To solve the quadratic inequality $a^2 - a - 1 \leq 0$, let us first solve the corresponding equality $a^2 - a - 1 = 0$. By the quadratic formula, there are two real solutions $a_1 = \frac{1-\sqrt{5}}{2}$ and $a_2 = \frac{1+\sqrt{5}}{2}$.

Since the parabola $f(a) = a^2 - a - 1$ opens upward, $a^2 - a - 1 \leq 0$ holds precisely when a is in the closed interval $[a_1, a_2]$, i.e. when

$$\frac{1-\sqrt{5}}{2} \leq a \leq \frac{1+\sqrt{5}}{2}.$$

Observe that $a = \frac{3}{2}$ is in this interval. We are looking for the largest possible a value, which is the right endpoint:

$$a = a_2 = \frac{1+\sqrt{5}}{2}$$

This quantity is famous as the golden ratio, or phi (φ). It is approximately equal to 1.618.

An Improved Estimate for the Fibonacci Sequence (3)

Thus, the largest number a for which our proof works is

$$a = \varphi = \frac{1+\sqrt{5}}{2}.$$

The student is encouraged to write the full inductive proof for that case, and to confirm that with $a = \varphi$, the last inequality in the inductive step becomes an equality.

With that, we have the following improved estimate for the Fibonacci sequence:

$$f_n \geq \left(\frac{1+\sqrt{5}}{2}\right)^{n-1}$$

for all $n \geq 0$.