

**PROFESSOR:** So whenever you define a recursive data type, implicit in the definition is a method for proving things about it called structural induction. And the way structural induction works is that if you want to prove that every element in some recursively defined data type has a particular property  $P$ , then you proceed by showing that every one of the elements in the base case that are in  $R$  has property  $P$ , and moreover that if you apply a constructor to an element  $x$ , then it has property  $P$  whenever  $x$  has property  $P$ . That is, you can assume as a structural induction hypothesis  $P$  of  $x$  and then you need to show that  $P$  of  $c$  of  $x$  holds.

And this applies for each constructor  $c$ . Some constructors take more than one argument, but this is meant to illustrate the general pattern. Let's do an easy example first. This is what we've actually seen. And we took for granted this method of proof without highlighting it when we argued that the set  $E$  that was recursively defined in the last presentation contained only even numbers.

So remember the definition of  $E$  was that the  $0$  is in  $E$ . And we're going to be proving that  $x$  is even by induction. So we need to check the base case, yes,  $0$  is even. And then we need to show that assuming the structural induction hypothesis that  $n$  is even, then when we apply the constructor  $n + 2$ , it's even-- well, obviously it is-- or if we apply the constructor minus  $n$ , that's also even.

And it is as well. And that's why structural induction tells us that in fact, every string in the set  $E$  is even. Now let's look at a somewhat more interesting example, which was the set  $M$  of matching right and left brackets. And what I want to prove by structural induction is that every string in  $M$  has the same number of left brackets and right brackets.

I can restate this by defining  $EQ$  to be the set of strings with the same number of right and left brackets. And what I'm really trying to say is that  $M$  is a subset of  $EQ$ . All right. Now the way I'm going to prove this is by defining my induction hypothesis  $P$  of  $s$  to be that  $s$  is in  $EQ$ , that is  $s$  has an equal number of left and right brackets.

Well, let's remember what the definition of  $M$  was. The base case of  $M$  was the empty string with no brackets at all. And does the empty string satisfy  $P$  of  $s$ ? Well, yeah. It has  $0$  right brackets and  $0$  left brackets so it does have an equal number of left and right brackets. So we've established that the base case  $P$  of the empty string is true.

Now we have to consider the constructor case. In the case of  $M$ , there's only one constructor, namely if  $r$  and  $t$  are in  $M$ , then so is  $s$ , which you get by putting brackets around  $r$  and following it by  $t$ . Well, here's the argument. We're allowed to assume-- we're trying to prove that  $s$  has an equal number of left and right brackets. And we're allowed to assume that  $r$  does and so does  $t$ .

So let's look at the number of right brackets in  $s$ . Well, where they come from? The right brackets in  $s$  consist of-- well, the first symbol in  $s$  is a left bracket, so that doesn't matter. Then it's the right brackets in  $r$ . And then there is a new right bracket that gets added. And then there are the right brackets in  $t$ .

So what I can say is that the right brackets in  $s$  are simply, the number of them is the sum of the number in  $r$  plus the number in  $t$  plus one more, cause the constructor threw in one more right bracket. By exactly the same reasoning, the number of left brackets in  $s$  is the number of left in  $r$ , left in  $t$  plus 1.

Now, because of hypothesis  $P$  of  $r$ , the number of right and left brackets in  $r$  are equal. And likewise, by the induction hypothesis  $P$  of  $t$ , the number of right and left brackets in  $t$  are equal. And so the right hand sides of both of these equations are equal. And that means that the left hand sides are equal. We've just proved that the number of right brackets in  $s$  and the number of left brackets in  $s$  are the same, so  $P$  of  $s$  is true. The constructor case is covered.

And we can conclude by structural induction that every  $s$  in the set  $M$ , recursively defined set of strings of matched brackets, is in fact has an equal number of left and right brackets, which means that  $M$  is a subset of  $Q$  as claimed.

Well, those were pretty easy structural inductions. And as with regular induction proofs, when you get the right induction hypothesis, the proofs tend to be easy. And we are going to work on an interesting example having to do with the F18 functions.

One of the reasons why the F18s are what's considered in first term calculus is that if you look at all of those functions-- remember, you've got them by taking constant functions and the identity function and the function sine  $x$ , then you could combine them in various ways by adding, multiplying exponentiating, composing, taking inverses-- that we didn't need to add a constructor of taking the derivative. Because it turns out that you can prove by structural induction that the F18s are closed under taking derivatives. And that makes a lovely class

problem.