50003 - Models of Computation - Lecture 5

Oliver Killane

25/10/21

Lecture Recording
Lecture recording is available here

Structural Induction

Structural induction is used for reasoning about collections of objects, which are:

- structured in a well defined way
- finite but can be arbitrarily large and complex

We can use this is reason about:

- natural numbers
- data structures (lists, trees, etc)
- programs (can be large, but are finite)
- derivations of assertions like $E \Downarrow 4$ (finite trees of axioms and rules)

Structural Induction over Natural Numbers

$$\mathbb{N} \in Nat ::= zero | succ(\mathbb{N})$$

To prove a property $P(\mathbb{N})$ holds, for every number $N \in Nat$ by induction on structure $\mathbb{N}$:

- **Base Case** Prove $P(zero)$
- **Inductive Case** Inductive Case is $P(Succ(K))$ where $P(K)$ holds

For example, we can prove the property:

$$plus(\mathbb{N}, zero) = \mathbb{N}$$

- **Base Case**
  Show $plus(zero, zero) = zero$

$$\begin{array}{llll} (1) & \text{LHS} & = & plus(zero, zero) & \\ (2) & & = & zero & \text{(By definition of } plus\text{)} \\ (3) & & = & \text{RHS} & \text{(As Required)} \end{array}$$

- **Inductive Case**
  $N = succ(K)$
  Inductive Hypothesis $plus(K, zero) = K$
  Show $plus(succ(K), zero) = succ(K)$

$$\begin{array}{lllll} (1) & \text{LHS} & = & plus(succ(K), zero) & \\ (2) & & = & succ(plus(K, zero)) & \text{(By definition of } plus\text{)} \\ (3) & & = & succ(K) & \text{(By Inductive Hypothesis)} \\ (4) & & = & \text{RHS} & \text{(As Required)} \end{array}$$

Mathematics induction is a special case of structural induction:

$$P(0) \wedge [\forall k \in \mathbb{N}. P(k) \Rightarrow P(k+1)]$$

In the exam you may use $P(0)$ and $P(K+1)$ rather than $P(zero)$ and $P(succ(k))$ to save time.

Binary Tree Example

$$bTree \in BinaryTree ::= Node|Branch(bTree, bTree)$$

We can define a function *leaves*:

$$leaves(Node) = 1$$

$$leaves(Branch(T_1, T_2)) = 1 + leaves(T_1) + leaves(T_2)$$

Or *branches*:

$$branches(Node) = 0$$

$$branches(Branch(T_1, T_2)) = branches(T_1) + branches(T_2)$$

Exercise

Prove By induction that $leaves(T) = branches(T) + 1$

Induction over SimpleExp

$$E \in SimpleExp ::= n|E + E|E \times E| \dots$$

where $n \in N$.

Properties of $\Downarrow$

- **Determinacy**
  A simple expression can only evaluate to one answer.

$$E \Downarrow n_1 \wedge E \Downarrow n_2 \rightarrow n_1 = n_2$$

- **Totality**
  A simple expression evaluates to at least one answer.

$$\forall E \in SimpleExp . \exists n \in \mathbb{N} . [E \Downarrow n]$$