Introduction to Recursion and Induction

Alan Bundy
School of informatics
University of Edinburgh

Mathematical Induction Experiment

- Purpose: relation between formal and informal understanding.
- Short introduction to induction and recursion.
- Group exercise about inductive conjectures.

The Need for Mathematical Induction

Proof by mathematical induction required for reasoning about repetition, *e.g.* in:

- recursive datatypes,

  numbers, lists, sets, trees, *etc*;
- iterative or recursive computer programs;
- electronic circuits with loops or parameterisation;
- theorems in number theory, e.g. Fermat's Last Theorem.

Structure of an Induction Rule

$$\frac{P(0), \quad \forall n{:}nat.\ (P(n) \rightarrow P(n+1))}{\forall n{:}nat.\ P(n)}$$

Base Case: $P(0)$

Step Case: $\forall n{:}nat.\ (P(n) \rightarrow P(n+1))$

Induction Variable: n

Induction Term: $n+1$

Induction Hypothesis: $P(n)$

Induction Conclusion: $P(n+1)$

Sequent Form: $P(n) \vdash P(n+1)$

Recursive Datatypes

Examples: natural numbers, lists, sets, trees.

Made From: base and step constructor functions.

Naturals: $0 : nat$ and $s : nat \mapsto nat$.

$$nat = \{0, s(0), s(s(0)), s(s(s(0))), \ldots\}$$

Lists: $[\,] : list(\tau)$ and $[\ldots|\ldots] \; : \tau \times list(\tau) \mapsto list(\tau)$.

$$list(\tau) = \{[\,], [\alpha_1], [\alpha_2, \alpha_1], \ldots\}$$

where $\alpha_i : \tau$.

Recursive Definition

Example: addition on naturals.

$$0 + Y = Y$$
$$s(X) + Y = s(X + Y)$$

Base Case: $0 + Y = Y$

Step Case: $s(X) + Y = s(X + Y)$

Recursion Variable: X

More Examples of Recursion

Two Step: even on naturals.

$$
\begin{aligned}
even(0) &\leftrightarrow \mathbf{true} \\
even(s(0)) &\leftrightarrow \mathbf{false} \\
even(s(s(n))) &\leftrightarrow even(n)
\end{aligned}
$$

Lists: append of two lists.

$$
\begin{aligned}
[\,] <> L &= L \\
([H|T]) <> L &= [H|(T <> L)]
\end{aligned}
$$

Rewriting

Rewrite Rule of Inference:

$$\frac{Cond \rightarrow lhs \Rightarrow rhs, \quad P[rhs\phi], \quad Cond\phi}{P[sub]}$$

where $lhs\phi \equiv sub$,

often $Cond$ is absent.

Example Rewriting:

$$\frac{2 \times X \Rightarrow X + X, \quad even(n + n)}{even(2 \times n)}$$

Example Rewrite Rules

From Recursive Definitions: *e.g.* of addition.

$$0 + Y \Rightarrow Y$$

$$s(X) + Y \Rightarrow s(X + Y)$$

Conditional Rewrite Rule:

$$Y \neq 0 \rightarrow X \Rightarrow quot(X, Y) \times Y + rem(X, Y)$$

Lemma as Rewrite Rule:

$$X + s(Y) \Rightarrow s(X + Y)$$

Non-Termination Problem: *e.g.* commutativity law.

$$X + Y \Rightarrow Y + X$$

Example Inductive Proof

Conjecture: the associativity of append.

$$\forall x\!:\!list(\tau)\forall y\!:\!list(\tau)\forall z\!:\!list(\tau).$$
$$x <> (y <> z) = (x <> y) <> z$$

Rewrite Rules: from definition of append.

$$[\,] <> L \Rightarrow L$$
$$([H|T]) <> L \Rightarrow [H|(T <> L)]$$

Induction Rule: one-step on lists.

$$\frac{P([\,]) \quad \forall h\!:\!\tau.\forall t\!:\!list(\tau).\ P(t) \rightarrow P([h|t])}{\forall l\!:\!list(\tau).\ P(l)}$$

Base Case Proof

Base Case: induction on x.

$$[\,] <> (y <> z) = ([\,] <> y) <> z$$

Rewriting Steps:

$$y <> z = ([\,] <> y) <> z$$

$$y <> z = y <> z$$

Rewrite Rule:

$$[\,] <> L \Rightarrow L$$

Step Case Proof

Step Case: induction on x.

$$t <> (Y <> Z) = (t <> Y) <> Z$$
$$\vdash ([h|t]) <> (y <> z) = (([h|t]) <> y) <> z$$

Rewriting Steps:

$$[h|(t <> (y <> z))] \quad = \quad ([h|(t <> y))] <> z$$
$$[h|(t <> (y <> z))] \quad = \quad [h|((t <> y) <> z)]$$
$$h = h \wedge t <> (y <> z) = (t <> y) <> z$$

Rewrite Rules:

$$([H|T]) <> L \Rightarrow [H(T <> L)]$$
$$([H_1|T_1] = [H_2|T_2]) \Rightarrow (H_1 = H_2 \wedge T_1 = T_2)$$

Experiment

- Organise yourselves into groups of three (or four).
- Discuss the exercises in the handout with the rest of your group.
- Collectively solve these exercises.
- If you need help, raise your hand.
- Please record your intermediate working.

  Note down any thoughts, including any false starts.
- Choose a group spokesperson to explain your ideas to the rest of the class.