ENE4014: Programming Languages

Lecture 2 — Inductive Definitions (2)

Woosuk Lee
2024 Spring

Contents

- More examples of inductive definitions
  - natural numbers, strings, booleans
  - lists, binary trees
  - arithmetic expressions, propositional logic
- Structural induction
  - three example proofs

Natural Numbers

The set of natural numbers:

$$\mathbb{N} = \{0, 1, 2, 3, \ldots\}$$

is inductively defined:

$$\frac{}{0} \qquad \frac{n}{n+1}$$

The inference rules can be expressed by a grammar:

$$n \to 0 \mid n + 1$$

Interpretation:

- 0 is a natural number.
- If n is a natural number then so is $n + 1$.

Strings

The set of strings over alphabet $\{\mathtt{a}, \ldots, \mathtt{z}\}$, e.g., ϵ, a, b, . . . , z, aa, ab, . . . , az, ba, . . . az, aaa, . . . , zzz, and so on. Inference rules:

$$\frac{}{\boldsymbol{\epsilon}} \qquad \frac{\boldsymbol{\alpha}}{\mathtt{a}\boldsymbol{\alpha}} \qquad \frac{\boldsymbol{\alpha}}{\mathtt{b}\boldsymbol{\alpha}} \quad \cdots \qquad \frac{\boldsymbol{\alpha}}{\mathtt{z}\boldsymbol{\alpha}}$$

or simply,

$$\frac{}{\boldsymbol{\epsilon}} \qquad \frac{\boldsymbol{\alpha}}{\boldsymbol{x\alpha}}\ \boldsymbol{x} \in \{\mathtt{a}, \ldots, \mathtt{z}\}$$

In grammar:

$$\begin{array}{rcll} \boldsymbol{\alpha} & \rightarrow & \boldsymbol{\epsilon} & \\ & \mid & \boldsymbol{x\alpha} & (\boldsymbol{x} \in \{\mathtt{a}, \ldots, \mathtt{z}\}) \end{array}$$

Boolean Values

The set of boolean values:

$$\mathbb{B} = \{\boldsymbol{true}, \boldsymbol{false}\}.$$

If a set is finite, just enumerate all of its elements by axioms:

$$\frac{}{\boldsymbol{true}} \qquad \frac{}{\boldsymbol{false}}$$

In grammar:

$$\boldsymbol{b} \rightarrow \boldsymbol{true} \mid \boldsymbol{false}$$

Lists

Examples of lists of integers:

1. $\mathbf{nil}$
2. $\mathbf{14 \cdot nil}$
3. $\mathbf{3 \cdot 14 \cdot nil}$
4. $\mathbf{-7 \cdot 3 \cdot 14 \cdot nil}$

Inference rules:

$$\frac{}{\mathbf{nil}} \qquad \frac{\boldsymbol{l}}{\boldsymbol{n \cdot l}}\ \boldsymbol{n \in \mathbb{Z}}$$

In grammar:

$$\begin{array}{rcll} \boldsymbol{l} & \rightarrow & \mathbf{nil} & \\ & \mid & \boldsymbol{n \cdot l} & (\boldsymbol{n \in \mathbb{Z}}) \end{array}$$

Lists

A proof that $\mathbf{-7}\cdot\mathbf{3}\cdot\mathbf{14}\cdot\textbf{nil}$ is a list of integers:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\textbf{nil}}}{\mathbf{14}\cdot\textbf{nil}}\ \mathbf{14}\in\mathbb{Z}}{\mathbf{3}\cdot\mathbf{14}\cdot\textbf{nil}}\ \mathbf{3}\in\mathbb{Z}}{\mathbf{-7}\cdot\mathbf{3}\cdot\mathbf{14}\cdot\textbf{nil}}\ \mathbf{-7}\in\mathbb{Z}$$

The proof tree is also called *derivation tree* or *deduction tree*.

Binary Trees

Examples of binary trees:

1. $\textbf{leaf}$
2. $(2, \textbf{leaf}, \textbf{leaf})$
3. $(1, (2, \textbf{leaf}, \textbf{leaf}), \textbf{leaf})$
4. $(1, (2, \textbf{leaf}, \textbf{leaf}), (3, (4, \textbf{leaf}, \textbf{leaf}), \textbf{leaf}))$

Inference rules:

$$\frac{}{\textbf{leaf}} \qquad \frac{t_1 \quad t_2}{(n, t_1, t_2)}\; n \in \mathbb{Z}$$

In grammar:

$$\begin{array}{rcl} t & \rightarrow & \textbf{leaf} \\ & \mid & (n, t, t) \quad (n \in \mathbb{Z}) \end{array}$$

Binary Trees

A proof that

$$(\mathbf{1}, (\mathbf{2}, \textbf{leaf}, \textbf{leaf}), (\mathbf{3}, (\mathbf{4}, \textbf{leaf}, \textbf{leaf}), \textbf{leaf}))$$

is a binary tree:

$$\dfrac{\dfrac{\dfrac{}{\textbf{leaf}}}{(\mathbf{2}, \textbf{leaf}, \textbf{leaf})}\ \mathbf{2} \in \mathbb{Z} \quad \dfrac{\dfrac{\dfrac{}{\textbf{leaf}}}{(\mathbf{4}, \textbf{leaf}, \textbf{leaf})}\ \mathbf{4} \in \mathbb{Z}}{(\mathbf{3}, (\mathbf{4}, \textbf{leaf}, \textbf{leaf}), \textbf{leaf})}\ \mathbf{3} \in \mathbb{Z}}{(\mathbf{1}, (\mathbf{2}, \textbf{leaf}, \textbf{leaf}), (\mathbf{3}, (\mathbf{4}, \textbf{leaf}, \textbf{leaf}), \textbf{leaf}))}\ \mathbf{1} \in \mathbb{Z}$$

Binary Trees: a different version

Binary tree examples: $\mathbf{1}$, $(\mathbf{1}, \textbf{nil})$, $(\mathbf{1}, \mathbf{2})$, $((\mathbf{1}, \mathbf{2}), \textbf{nil})$, $((\mathbf{1}, \mathbf{2}), (\mathbf{3}, \mathbf{4}))$.
Inference rules:

$$\frac{}{\boldsymbol{n}}\ \boldsymbol{n} \in \mathbb{Z} \qquad \frac{\boldsymbol{t}}{(\boldsymbol{t}, \textbf{nil})} \qquad \frac{\boldsymbol{t}}{(\textbf{nil}, \boldsymbol{t})} \qquad \frac{\boldsymbol{t_1} \qquad \boldsymbol{t_2}}{(\boldsymbol{t_1}, \boldsymbol{t_2})}$$

In grammar:

$$\begin{array}{rcll} \boldsymbol{t} & \rightarrow & \boldsymbol{n} & (\boldsymbol{n} \in \mathbb{Z}) \\ & \mid & (\boldsymbol{t}, \textbf{nil}) & \\ & \mid & (\textbf{nil}, \boldsymbol{t}) & \\ & \mid & (\boldsymbol{t}, \boldsymbol{t}) & \end{array}$$

A proof that $((\mathbf{1}, \mathbf{2}), (\mathbf{3}, \textbf{nil}))$ is a binary tree:

$$\frac{\dfrac{\dfrac{}{\mathbf{1}} \qquad \dfrac{}{\mathbf{2}}}{(\mathbf{1}, \mathbf{2})} \qquad \dfrac{\dfrac{}{\mathbf{3}}}{(\mathbf{3}, \textbf{nil})}}{((\mathbf{1}, \mathbf{2}), (\mathbf{3}, \textbf{nil}))}$$

Expressions

Expression examples: $\mathbf{2}$, $-\mathbf{2}$, $\mathbf{1}+\mathbf{2}$, $\mathbf{1}+(\mathbf{2}*(-\mathbf{3}))$, etc.
Inference rules:

$$\frac{}{\boldsymbol{n}}\ \boldsymbol{n}\in\mathbb{Z} \qquad \frac{\boldsymbol{e}}{-\boldsymbol{e}} \qquad \frac{\boldsymbol{e_1} \quad \boldsymbol{e_2}}{\boldsymbol{e_1}+\boldsymbol{e_2}} \qquad \frac{\boldsymbol{e_1} \quad \boldsymbol{e_2}}{\boldsymbol{e_1}*\boldsymbol{e_2}} \qquad \frac{\boldsymbol{e}}{(\boldsymbol{e})}$$

In grammar:

$$\begin{array}{rcll} \boldsymbol{e} & \to & \boldsymbol{n} & (\boldsymbol{n}\in\mathbb{Z}) \\ & \mid & -\boldsymbol{e} & \\ & \mid & \boldsymbol{e}+\boldsymbol{e} & \\ & \mid & \boldsymbol{e}*\boldsymbol{e} & \\ & \mid & (\boldsymbol{e}) & \end{array}$$

Example:

$$\dfrac{\dfrac{}{\mathbf{1}} \quad \dfrac{\dfrac{\dfrac{}{\mathbf{2}} \quad \dfrac{\dfrac{\dfrac{}{\mathbf{3}}}{-\mathbf{3}}}{(-\mathbf{3})}}{\mathbf{2}*(-\mathbf{3})}}{(\mathbf{2}*(-\mathbf{3}))}}{\mathbf{1}+(\mathbf{2}*(-\mathbf{3}))}$$

Propositional Logic

Examples:

- $\boldsymbol{T}$, $\boldsymbol{F}$
- $\boldsymbol{T} \wedge \boldsymbol{F}$
- $\boldsymbol{T} \vee \boldsymbol{F}$
- $(\boldsymbol{T} \wedge \boldsymbol{F}) \wedge (\boldsymbol{T} \vee \boldsymbol{F})$
- $\boldsymbol{T} \Rightarrow (\boldsymbol{F} \Rightarrow \boldsymbol{T})$

Propositional Logic

Syntax:

$$\begin{array}{rcl} f & \rightarrow & T \mid F \\ & \mid & \neg f \\ & \mid & f \wedge f \\ & \mid & f \vee f \\ & \mid & f \Rightarrow f \end{array}$$

Semantics ($[\![f]\!]$):

$$\begin{array}{rcl} [\![T]\!] & = & \mathit{true} \\ [\![F]\!] & = & \mathit{false} \\ [\![\neg f]\!] & = & \text{not } [\![f]\!] \\ [\![f_1 \wedge f_2]\!] & = & [\![f_1]\!] \text{ andalso } [\![f_2]\!] \\ [\![f_1 \vee f_2]\!] & = & [\![f_1]\!] \text{ orelse } [\![f_2]\!] \\ [\![f_1 \Rightarrow f_2]\!] & = & [\![f_1]\!] \text{ implies } [\![f_2]\!] \end{array}$$

Propositional Logic

$$\begin{aligned}
[\![(\boldsymbol{T} \land (\boldsymbol{T} \lor \boldsymbol{F})) \Rightarrow \boldsymbol{F}]\!] &= [\![\boldsymbol{T} \land (\boldsymbol{T} \lor \boldsymbol{F})]\!] \text{ implies } [\![\boldsymbol{F}]\!] \\
&= ([\![\boldsymbol{T}]\!] \text{ andalso } [\![\boldsymbol{T} \lor \boldsymbol{F}]\!]) \text{ implies } \boldsymbol{false} \\
&= (\boldsymbol{true} \text{ andalso } ([\![\boldsymbol{T}]\!] \text{ orelse } [\![\boldsymbol{F}]\!])) \text{ implies } \boldsymbol{false} \\
&= (\boldsymbol{true} \text{ andalso } (\boldsymbol{true} \text{ orelse } \boldsymbol{false})) \text{ implies } \boldsymbol{false} \\
&= \boldsymbol{false}
\end{aligned}$$

Structural Induction

A technique for proving properties about inductively defined sets.

To prove that a proposition $P(s)$ is true for all structures s, prove the following:

1. (Base case) P is true on simple structures (those without substructures)
2. (Inductive case) If P is true on the substructures of s, then it is true on s itself. The assumption is called *induction hypothesis (I.H.)*.

Example 1

Let S be the set defined by the following inference rules:

$$\frac{}{3} \qquad \frac{x \quad y}{x+y}$$

Prove that for all $x \in S$, x is divisible by 3.
Proof. By structural induction.

- (Base case) The base case is when x is 3. Obviously, x is divisible by 3.
- (Inductive case) The induction hypothesis (I.H.) is

  $$x \text{ is divisible by } 3, \qquad y \text{ is divisible by } 3.$$

  Let $x = 3k_1$ and $y = 3k_2$. Using I.H., we derive

  $$x + y \text{ is divisible by } 3$$

  as follows:

  $$\begin{array}{rcll} x + y & = & 3k_1 + 3k_2 & \cdots \text{by I.H.} \\ & = & 3(k_1 + k_2) & \end{array}$$

□

Example 2

Let S be the set defined by the following inference rules:

$$\frac{}{()} \qquad \frac{x}{(x)} \qquad \frac{x \qquad y}{xy}$$

Prove that every element of the set has the same number of ('s and)'s.
Proof Restate the claim formally:

$$\text{If } x \in S \text{ then } l(x) = r(x)$$

where $l(x)$ and $r(x)$ denote the number of ('s and)'s, respectively:

$$\begin{array}{rcl} l(()) & = & 1 \\ l((x)) & = & l(x) + 1 \\ l(xy) & = & l(x) + l(y) \end{array} \qquad \begin{array}{rcl} r(()) & = & 1 \\ r((x)) & = & r(x) + 1 \\ r(xy) & = & r(x) + r(y) \end{array}$$

We prove it by structural induction:

- (Base case): The base case is when $x = ()$. Then $l(x) = 1 = r(x)$.

Example 2

- (Inductive case): There are two inductive cases:

$$\frac{x}{(x)} \qquad \frac{x \qquad y}{xy}$$

  Induction hypotheses (I.H.):

$$l(x) = r(x), \qquad l(y) = r(y).$$

  - The first case. We prove $l((x)) = r((x))$:

$$\begin{array}{rcll} l((x)) & = & l(x) + 1 & \cdots \text{by definition of } l((x)) \\ & = & r(x) + 1 & \cdots \text{by I.H.} \\ & = & r((x)) & \cdots \text{by definition of } r((x)) \end{array}$$

  - The second case. We prove $l(xy) = r(xy)$:

$$\begin{array}{rcll} l(xy) & = & l(x) + l(y) & \cdots \text{by definition of } l(xy) \\ & = & r(x) + r(y) & \cdots \text{by I.H.} \\ & = & r(xy) & \cdots \text{by definition of } r(xy) \end{array}$$

□

Example 3

Let T be the set of binary trees:

$$\frac{}{\textbf{leaf}} \qquad \frac{t_1 \quad t_2}{(n, t_1, t_2)}\ n \in \mathbb{Z}$$

Prove that for all such trees, the number of leaves is always one more than the number of internal nodes.

Proof. Restate the claim more formally:

$$\text{If } t \in T \text{ then } l(t) = i(t) + 1$$

where $l(t)$ and $i(t)$ denote the number of leaves and internal nodes, respectively:

$$\begin{array}{rcl} l(\textbf{leaf}) & = & 1 \\ l(n, t_1, t_2) & = & l(t_1) + l(t_2) \end{array} \qquad \begin{array}{rcl} i(\textbf{leaf}) & = & 0 \\ i(n, t_1, t_2) & = & i(t_1) + i(t_2) + 1 \end{array}$$

We prove it by structural induction:

- (Base case): The base case is when $t = \textbf{leaf}$, where $l(t) = 1$ and $i(t) = 0$.
- (Inductive case): The induction hypothesis:

  $$l(t_1) = i(t_1) + 1, \qquad l(t_2) = i(t_2) + 1$$

  Using I.H., we prove $l((n, t_1, t_2)) = i((n, t_1, t_2)) + 1$:

  $$\begin{array}{rcll} l((n, t_1, t_2)) & = & l(t_1) + l(t_2) & \text{definition of of } l \\ & = & i(t_1) + 1 + i(t_2) + 1 & \text{by induction hypothesis} \\ & = & i(n, t_1, t_2) + 1 & \text{definition of } i \end{array}$$

Summary

- Computer science is full of inductive definitions.
  - primitive values: booleans, characters, integers, strings, etc
  - compound values: lists, trees, graphs, etc
  - language syntax and semantics
- Structural induction
  - a general technique for reasoning about inductively defined sets