Lecture 3

Last time: Type system to reject programs like
"Hello" ^ "World"

Type safety: "Well-typed programs can't go wrong."
Robin Milner

2 Components:
Progress: If e is well-typed, it's a value or can take a step.

Preservation: If a well-typed exp. takes a step, still well-typed w/ the same type.

\[ e_1 \Rightarrow e_2 \Rightarrow e_3 \Rightarrow \ldots \Rightarrow v \]

Preservation: If \( e : \tau \) and \( e \Rightarrow e' \) then \( e' : \tau \)

Pf: By induction on the derivation of \( e \Rightarrow e' \)

S-1 By **inversion on T-3**, \( \tau = \text{int} \).
   By T-1, \( \bar{n}_1 + n_2 : \text{int} \).
S-2 By **inversion on T-4**, \( \tau = \text{string} \).
   By T-2, \( \text{"s","t" : string} \).
S-3 By **inversion on T-5**, \( \tau = \text{int} \).
   By T-1, \( 1s : \text{int} \).
S-4 By **inversion on T-3**, \( \tau = \text{int} \).
   By T-1, \( e_1 : \text{int} \).
   By T-3, \( e_1 + e_2 : \text{int} \).
S-5 By **inversion on T-3**, \( \tau = \text{int} \).
   By T-1, \( e_2 : \text{int} \).
   By T-3, \( \bar{n}_1 + e_2 : \text{int} \).
S-6, S-7, S-8 similar to above.
Lemma: Canonical Forms

1. If \( e \) \( \text{val} \) and \( e : \text{int} \), then \( e = \overline{n} \) for some \( n \).
2. If \( e \) \( \text{val} \) and \( e : \text{string} \), then \( e = "s" \) for some \( s \).

Pf: The only rules that can derive \( e \) \( \text{val} \) are \( V-1 \) and \( V-2 \).
   If \( V-1 \), then \( e = \overline{n} \) and \( e : \text{int} \).
   If \( V-2 \), then \( e = "s" \) and \( e : \text{string} \). \( \square \)

Progress: If \( e : \text{int} \), then \( e \) \( \text{val} \) or there exists \( e' \) s.t. \( e = e' \).

Pf: By induction on the derivation of \( e : \text{int} \).

\( T-1 \) \( \overline{n} \) \( \text{val} \) by \( V-1 \)

\( T-2 \) 
\( "s" \) \( \text{val} \) by \( V-2 \)

\( T-3 \) Then \( e = \text{int} \), \( e = e_1 + e_2 \), \( e_1 : \text{int} \), and \( e_2 : \text{int} \).

By \( 1H \), \( e_1 \) \( \text{val} \) or \( e_1 \rightarrow e_1' \).

\* \( e_1 \) \( \text{val} \).

By \( CF \), \( e_1 = \overline{n_1} \) for some \( n_1 \).

By \( 1H \), \( e_2 \) \( \text{val} \) or \( e_2 \rightarrow e_2' \).

\* \( e_2 \) \( \text{val} \).

By \( CF \), \( e_2 = \overline{n_2} \) for some \( n_2 \).

By \( S + \), \( \overline{n_1 + n_2} \rightarrow \overline{n_1 + n_2} \).

\* \( e_2 \rightarrow e_2' \). By \( S - \), \( \overline{n_1 + n_2} \rightarrow \overline{n_1} + \overline{n_2} \).

\* \( e_1 \rightarrow e_1' \). By \( S + \), \( e_1 + e_2 \rightarrow e_1' + e_2' \).

\( T-4 \), \( T-5 \) similar to above. \( \square \)