Announcements

• HW6 due Fri at 11:55pm
• No new homework this week
  - Prepare for Midterm 2 next week
  - Review in recitation next Tue

One-Slide Summary

• If T is a subtype of U then any expression of type T can be used in a context that expects a U; this is called subsumption.
• A conversion is a function that converts between types.
• Widening conversions convert between types respecting subsumption, while narrowing is counter to subsumption.
• Subtyping rules for constructed types may be covariant, contravariant, or invariant.

Subtyping

\[
\text{If } T <: U \text{ and } e : T, \text{ then } e : U
\]

\[\text{rule}\]
Type checking subranges

- First, figure out which assignments we should allow.
  Second, discuss how we can use subtyping to check them.

VAR x : [1 TO 10];
VAR y : INTEGER;
x := 2;
y := 2;
x := 200;
y := 200;
x := y;
y := x;

Subtyping may break type safety

- With "bad" definitions of the subtyping relation, we may break type safety.

Plan for Safe Subtyping

- Define rules for base types based on subsets
  SHORT <: INTEGER [a TO b] <: INTEGER

- Define rules for type constructors that respect safety

Subtyping for Subranges

- [a TO b] <: [c TO d] if
  c <= a and  
  d >= b

Subtyping for Pairs

- T * T <: U * U' if
  T <: U  
  T' <: U'
  [0 TO 10] * INTEGER 0 (0, 55)  
  <:  
  [0 TO 100] * INTEGER
Subtyping for Pairs

- \(T \times T' \subseteq U \times U'\) if

Subtyping for Records

- \(\{l_1 : T_1, \ldots, l_n : T_n\} \subseteq \{l_1' : T_1', \ldots, l_n' : T_n'\}\)
  - if
    - \(T_i \subseteq T_i'\) and \(\ldots\) and \(T_n \subseteq T_n'\)
    - "depth subtyping for records"

Subtyping for Records

\[\begin{array}{c}
\text{Supersubclass} \\
\{x : \text{int}, y : \text{int}, z : \text{int}\} \subseteq \{x : \text{int}, y : \text{int}\}\end{array}\]

Subtyping for Records

- \(B \subseteq A\)

Easy? Subtyping for Functions

- \(T \rightarrow T' \subseteq U \rightarrow U'\) if \(T \subseteq U\) and \(T' \subseteq U'\)
  - Suppose \(f : [0 \text{ TO } 10] \rightarrow \text{INTEGER}\)
  - Claim: \(f(11)\) is "well-typed" with the above rule
  - How?
Subtyping for Functions

\[ T \to T' \subseteq U \to U' \text{ if } U \subseteq T \text{ and } T' \subseteq U' \]

One-Slide Summary

• If \( T \) is a \textit{subtype} of \( U \) then any expression of type \( T \) can be used in a context that expects a \( U \); this is called \textit{subsumption}.
• A \textit{conversion} is a function that converts between types.
• \textit{Widening conversions} convert between types respecting subsumption, while \textit{narrowing} is counter to subsumption.
• Subtyping rules for constructed types may be \textit{covariant}, \textit{contravariant}, or \textit{invariant}.

For Next Time

• Reading
• Online discussion forum
  - \( \geq 1 \) substantive question, comment, or answer each week
• Work on HW6
• Prepare for Midterm 2