

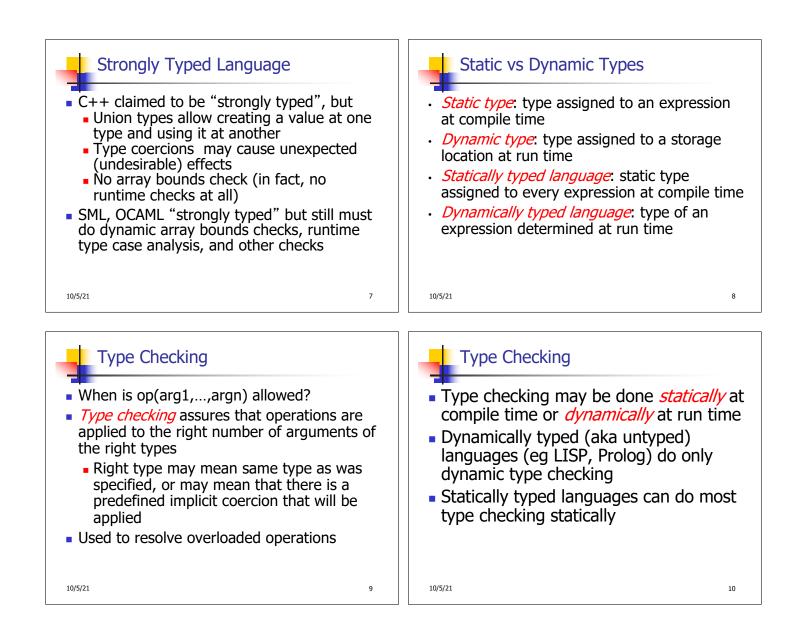
10/5/21

5

Most implementations of C and C++ do not

10/5/21

6



## Dynamic Type Checking

- Performed at run-time before each operation is applied
- Types of variables and operations left unspecified until run-time
  - Same variable may be used at different types

10/5/21

11

## Dynamic Type Checking

- Data object must contain type information
- Errors aren't detected until violating application is executed (maybe years after the code was written)

10/5/21

12

<ul> <li>Static Type Checking</li> <li>Performed after parsing, before code generation</li> <li>Type of every variable and signature of every operator must be known at compile time</li> </ul>	<ul> <li>Static Type Checking</li> <li>Can eliminate need to store type information in data object if no dynamic type checking is needed</li> <li>Catches many programming errors at earliest point</li> <li>Can't check types that depend on dynamically computed values</li> <li>Eg: array bounds</li> </ul>
10/5/21 13	10/5/21 14
<ul> <li>Static Type Checking</li> <li>Static Type Checking</li> <li>Stypically places restrictions on languages</li> <li>Garbage collection</li> <li>References instead of pointers</li> <li>All variables initialized when created</li> <li>Variable only used at one type</li> <li>Union types allow for work-arounds, but effectively introduce dynamic type checks</li> </ul>	<ul> <li>Type Declarations</li> <li>Type declarations: explicit assignment of types to variables (signatures to functions) in the code of a program</li> <li>Must be checked in a strongly typed language</li> <li>Often not necessary for strong typing or even static typing (depends on the type system)</li> </ul>
<ul> <li>Type Inference</li> <li>Type inference: A program analysis to assign a type to an expression from the program context of the expression</li> <li>Fully static type inference first introduced</li> </ul>	<ul> <li>Format of Type Judgments</li> <li>A <i>type judgement</i> has the form Γ  - exp : τ     </li> <li>Γ is a typing environment Supplies the types of variables (and function         </li> </ul>

- Fully static type inference first introduced by Robin Miller in ML
- Haskle, OCAML, SML all use type inference
  - Records are a problem for type inference

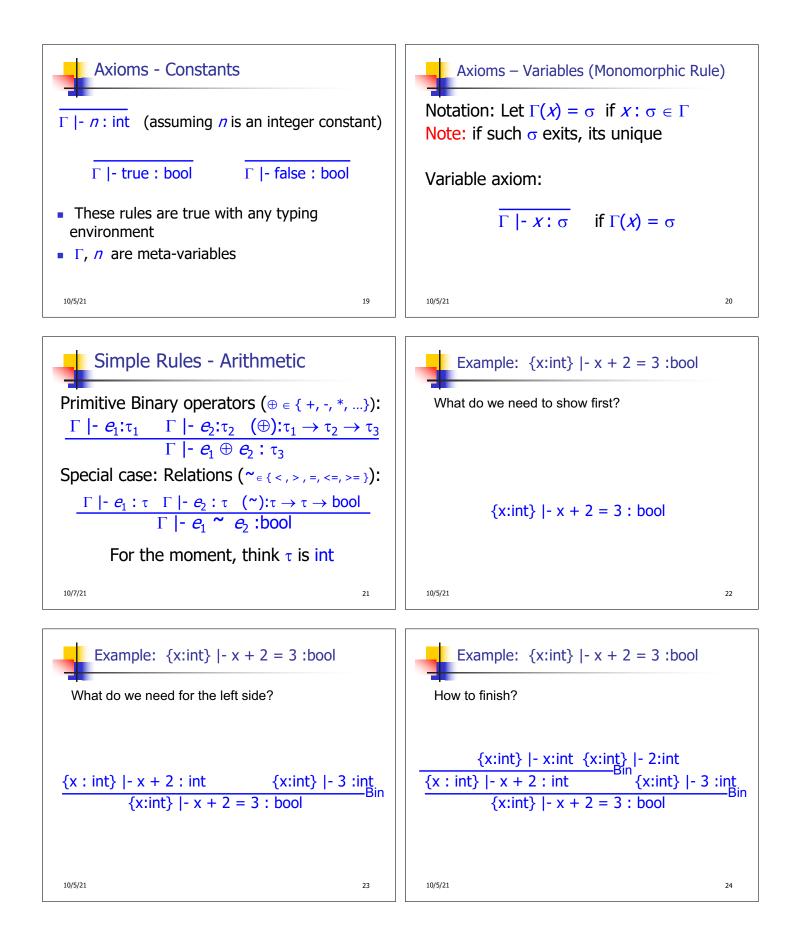
10/5/21

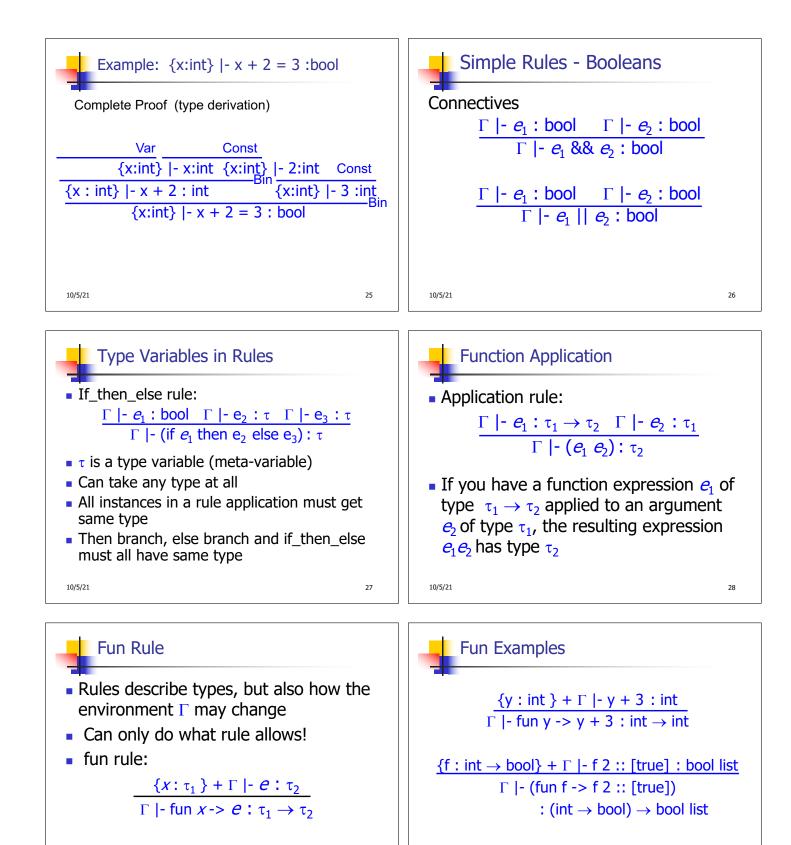
17

10/5/21

## Format of Type Judgments A *type judgement* has the form Γ |- exp : τ Γ is a typing environment Supplies the types of variables (and function names when function names are not variables) Γ is a set of the form { *x* : σ , . . . } For any *x* at most one σ such that (*x* : σ ∈ Γ) exp is a program expression τ is a type to be assigned to exp |- pronounced "turnstyle", or "entails" (or "satisfies" or, informally, "shows")

18





10/5/21

29

30

10/5/21

