Types

• A type consists of
  – a set of values
  – a set of operations on those values

• Types can be
  – primitive (atomic, non-decomposable)
  – composite (includes recursive types)

• Types can be
  – built-in
  – user-defined
Primitive types

• Typical primitive types
  – Boolean = \{true,false\}
  – Character = \{…,’a’,…’z’,…’0’,…,’9’,…\}
  – Integer = \{0, 1, -1, 2, -2, ... \}
  – Float = \{0.0, ..., 1.0, -1.0, ..., 3.1415, ...\}

• Set of primitive types is language-dependent

• Value of primitive types typically implementation-dependent, though can be written into language spec (e.g. Java, C#)
Gotcha’s

• Boolean is not always a distinct type
  – C++ bools are really ints

• Character is not always a distinct type
  – C, C++, Java are really just (unsigned) ints

• Languages sometimes provide many variants on a primitive type:
  – Java’s Integer types: byte, short, int, long