

## **Type Safety**

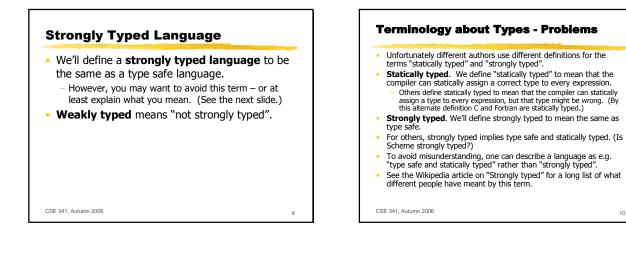
- A program is **type safe** if it is known to be free of type errors.
- However, the system is allowed to halt at runtime before performing an operation that would result in a type error.
- A language is type safe if all legal programs in that language are type safe.
- Java, Miranda, Smalltalk, Scheme, Haskell, and Ada are examples of type safe languages.
- Fortran and C are examples of languages that aren't type safe.
- Some languages for systems programming, for example Mesa, have a safe subset, although the language as a whole is not type safe.

CSE 341, Autumn 2006

## Tradeoffs

- Generally we want languages to be type safe.
- An exception is a language used for some kinds of systems programming, for example writing a garbage collector. The "safe subset" approach is one way to deal with this problem.
- Advantages of static typing: catch errors at compile time
  - machine-checkable documentation
  - potential for improved efficiency
- Advantages of dynamic typing:
- Flexibility
  - rapid prototyping

CSE 341, Autumn 2006



10