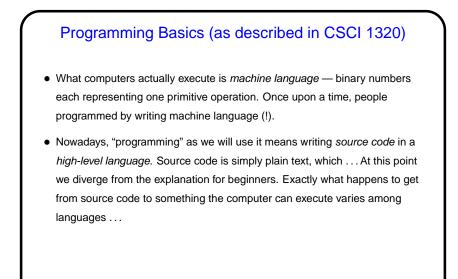
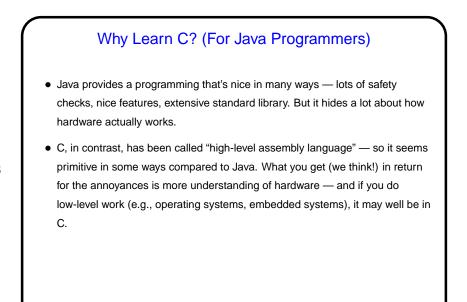


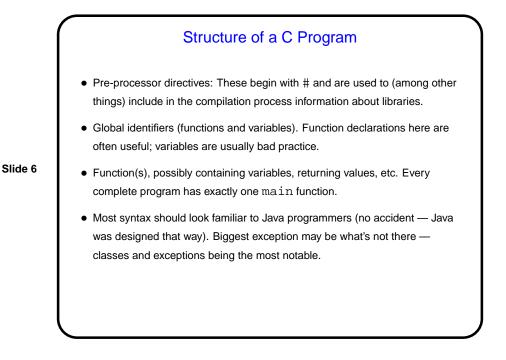
File Permissions in UNIX/Linux
Access to files specified in terms of three categories of users (owner, group, and other) and three kinds of access (read, write, and execute).
To show permissions, ls -l. First character says directory/not, then three groups of three letters each (rwx), one for each category of user.
To change permissions, chmod. Can specify via octal (base 8) numbers, but usually easier to use symbolic mode. Examples: chmod go= foo to say only owner can access foo. chmod go+r foo to say everyone can read foo (but not necessarily write it).



From Source Code to — What?

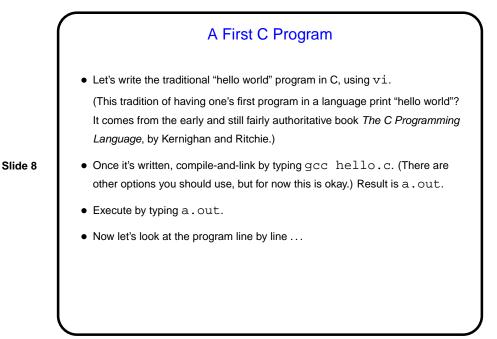
- Some high-level languages (such as the language understood by typical UNIX/Linux command shells) are directly interpreted by some other program.
- Others are *compiled* into *object code* (machine language) and then *linked* with other object code (including system libraries) to form an *executable* (something the operating system can execute).
- Java takes a somewhat intermediate approach it's initially compiled into byte code (object code for a made-up processor), which is (in principle) interpreted by the runtime system (Java Virtual Machine), with system library code brought in at runtime. (In practice, often a "just-in-time" compiler translates byte code into native object code on the fly.)

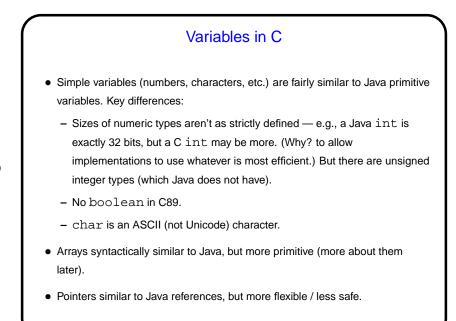




A Few Words About "Old C" Versus "New C"

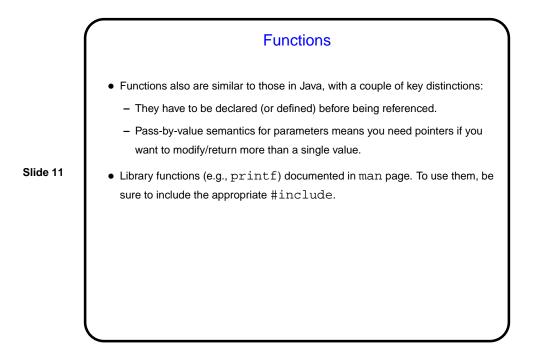
- First ANSI standard for C 1989. Widely adopted, but has some annoying limitations.
- Later standard 1999. Many features are widely implemented, but few compilers support the full standard, and older programs (and some programmers concerned about maximum portability) don't use new features. What we do in this class will focus on older standard for this reason.







- Most syntax is similar to Java (which is no accident) within each function, code is organized into statements, which may contain expressions.
- Control structures are mostly the same as in Java if , while, do, switch, for, etc. C doesn't have the simpler/newer form of for (referred to as "foreach").
- Key difference is the lack of classes (and supporting syntax), and use of pointers rather than references.



Minute Essay • TBA Slide 12