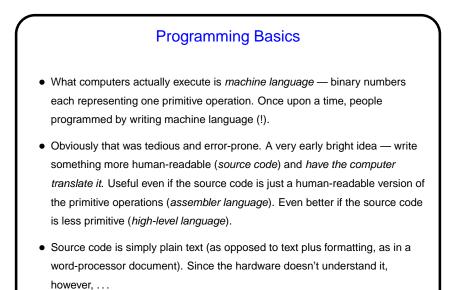


## Solving Problems on a Computer, Continued

- The overall process understand the problem, develop and test a solution is mostly independent of the choice of programming language and platform (combination of hardware and operating system, roughly). So once you understand the principles it is relatively easy to learn new languages.
- Opinions about which language to learn first, and on what platform, vary. For this course we will use Scala; it is somewhat easier for beginners than some of the other choices and also powerful enough to write interesting programs. We will also do most work from the command line under Linux.

Slide 3



**Programming Basics, Continued** 

Source code can be *interpreted* — translated line by line into something the hardware can understand, by another program called an *interpreter*.
 (This is how "scripting languages" work. An example is the command shell's language. !)

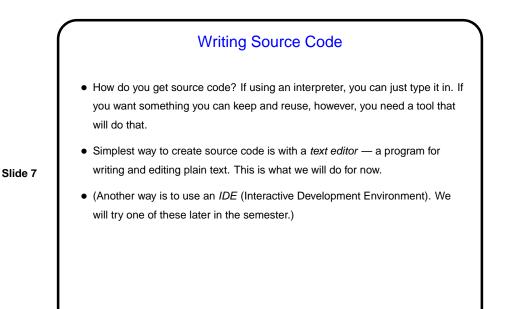
Slide 6

• Or it can be *compiled* — translated by a program called a *compiler* into something the hardware can execute directly.

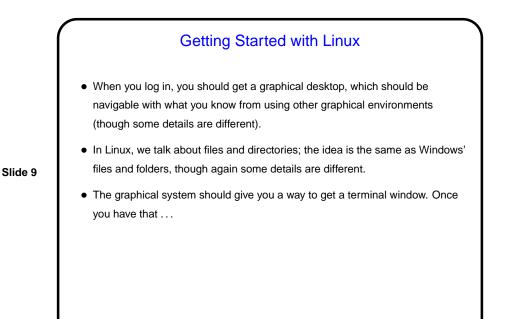
(This is how traditional "high-level" languages such as C and Fortran work.)

• Or it can be compiled into some intermediate form that can be executed by another program.

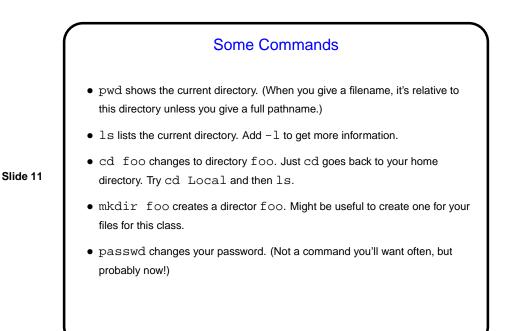
(This is how some recent languages such as Java work.)

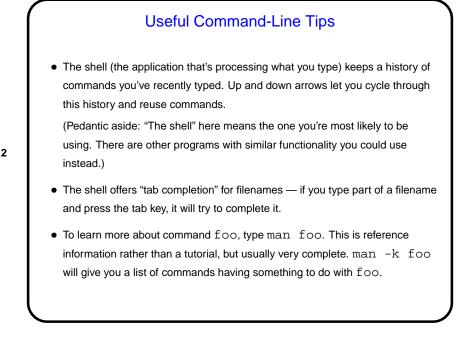


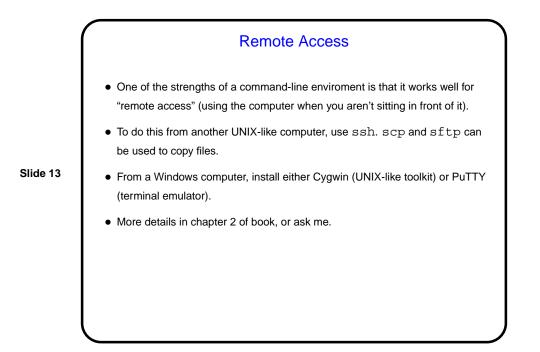
A Word About Tools • In this class we use Linux and command-line tools because we believe it is important for budding computer scientists to know how to work with these tools. For others — exposure to something new and different? Slide 8 • (What is Linux? it's an operating system, as Windows and Mac OS X are operating systems. It's one of a family of operating systems descended from UNIX, developed at Bell Labs in the early 1970s. A lot of servers run Linux or some other UNIX-like system. There are also ongoing efforts to develop mainstream desktop systems.) • A UNIX person's response to claims that UNIX isn't user-friendly: "Sure it is. It's just choosy about its friends."

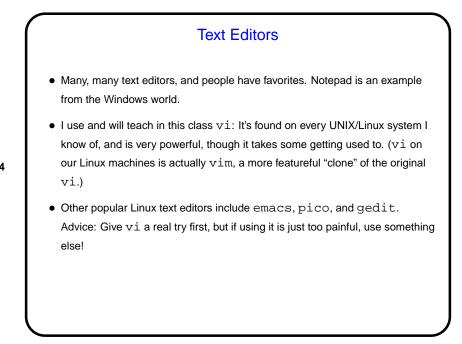


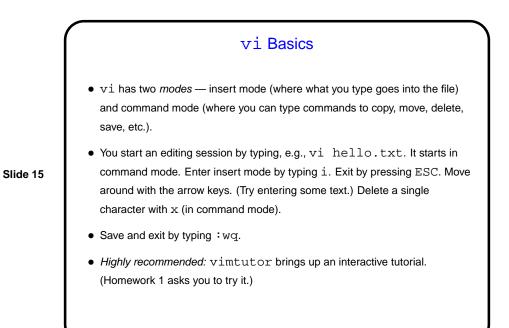
Getting Started With the Command Line
What you get when you start a terminal window is a "command shell", similar to Windows' "MS-DOS prompt".
Rather than pointing and clicking, you type the name of the program you want to run, plus whatever arguments (parameters) it needs.
(Why would you want to use a command line? because for some things it's arguably more efficient, and it's "scriptable" in ways that GUIs typically aren't.)
Let's try some commands ... (Don't worry if this goes by quickly — you should plan anyway to spend some time outside class trying out what we do in class and what's in the reading.)











vi Tips
Biggest hurdle may be the notion of modes. (But you already know about this, sort of? Word processors have insert/overwrite modes.)
Cut/copy/paste basics:

dd cuts a whole line. yy copies a whole line.
p pastes after the current line. P pastes before the current line.

Search by typing , text to search for, Enter. Repeat search with n. Search-and-replace using this, cw, and . (See book.)

