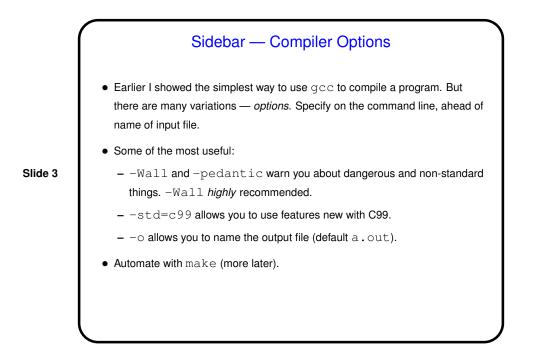
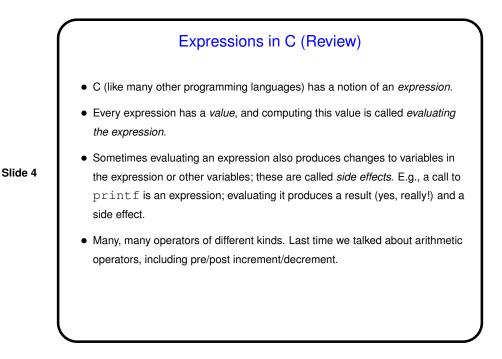


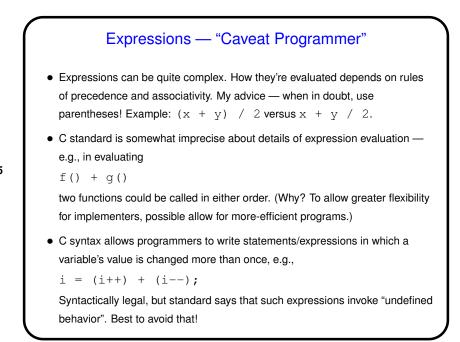
A Few Words About "Old C" Versus "New C"

- First ANSI standard for C 1989 ("C89"). Widely adopted, but has some annoying limitations.
- Later standard 1999 ("C99"). 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. Much of what we do in this class will focus on older standard for this reason.
 Some additions will work in gcc only with -std=c99 option.
- Still-later standard (2011) exists but is not (yet?) widely implemented.





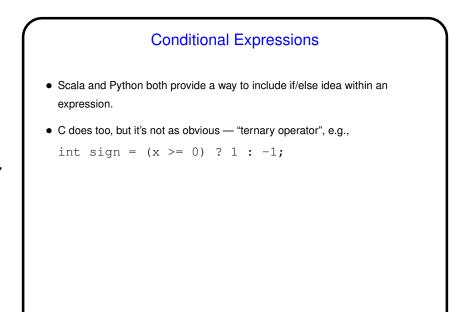




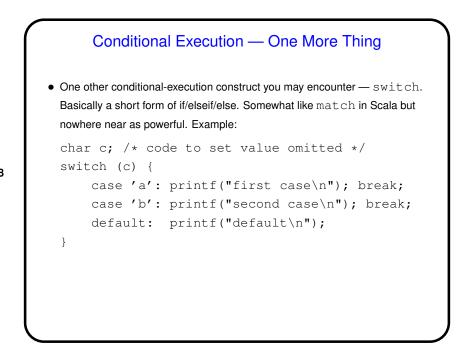
Slide 5

	Conditional Execution
•	As in other procedural languages, C has syntax for saying that some code should be executed only if some condition holds.
•	Syntax is
	if (boolean-expression)
	statement1
	else
	statement2
	where <i>statement1</i> and <i>statement2</i> can be single statements or blocks enclosed in curly braces.
	You can build up chains of conditions by making the statement after $else$
	another if, and you can omit the <code>else</code> and following statement. (The ideas
	here should be very familiar, and for most of you even the syntax should be
	pretty much what you know.)

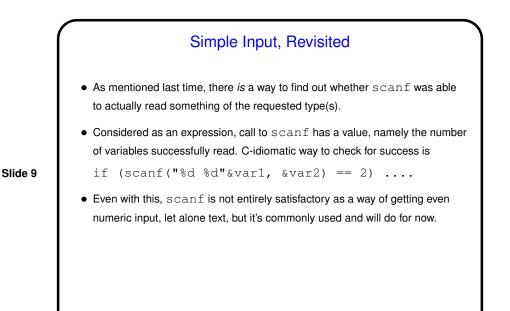
Slide 6



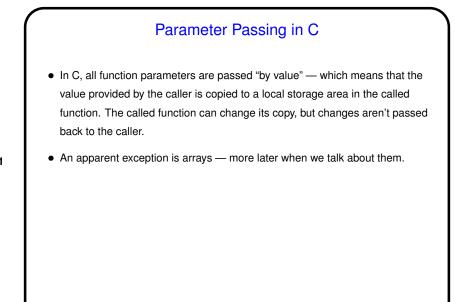
Slide 7



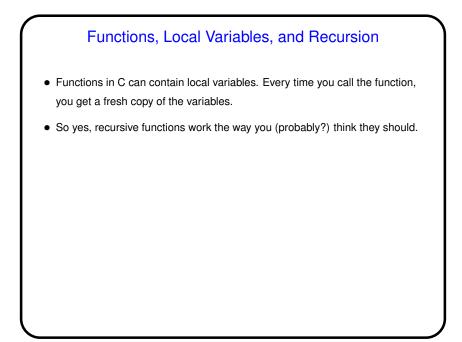
Slide 8



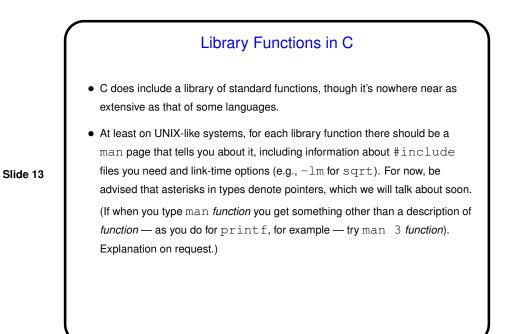
Functions in C
Functions in C are conceptually much like functions in other procedural programming languages. (Methods in object-oriented languages are similar but have some extra capabilities.)
I.e., a function has a *name*, *parameters*, a *return type*, and a *body* (some code).
Slide 10
One difference between C and higher-level languages: You aren't supposed to use a function before you tell the compiler about it, either by giving its full *definition* or by giving a *declaration* that specifies its name, parameters, and return type. The function body can be later in the same file or in some other file.
Also, C functions are not supposed to be nested (though some compilers allow it).



Slide 11



Slide 12



Minute Essay • What (if anything!) was interesting or difficult or otherwise noteworthy about Homework 2? Slide 14