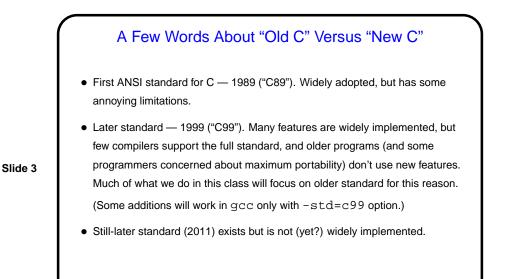
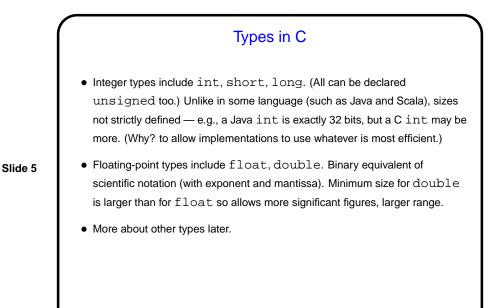


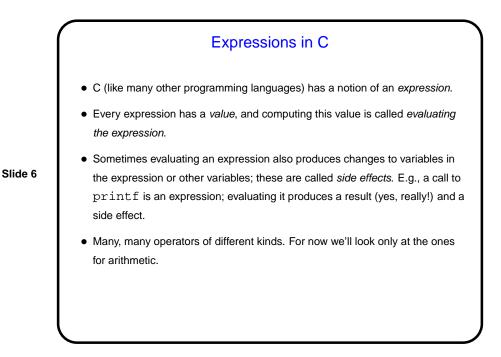
C Basics — Quick Overview Unlike Python and Scala scripts (but like Java programs), C programs include some standard boilerplate (#include for library functions, explicit main). Variables must be explicitly declared, including type. Expressions similar to those in Python/Scala/Java but with a few differences. Statements are also similar, but assignments are considered to be expressions too, with a value. Allows chaining, e.g., a = b = 10; A caveat: The C standard does not spell out everything (e.g., size of int type) so experimental results are not necessarily conclusive (might be specific to a particular compiler/system).

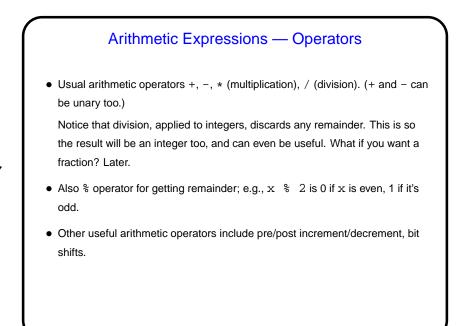
Slide 2



Variables in C
To do anything interesting in a program, we need some place to store input and intermediate values — "variables".
In C, variables must be *declared*, with a *name* and a *type*. (Contrast with Python, Scala.) In C89, all declarations must come before any code.
Variable names follow rules for *identifiers* — letters, numbers, and underscores only, must start with letter or underscore, preferably letter. Case-sensitive.
Is there anything like Scala's val versus var? Not exactly. Variables with const modifier cannot be directly assigned new values, but there are ways to evade this restriction using pointers. (More about pointers later.)

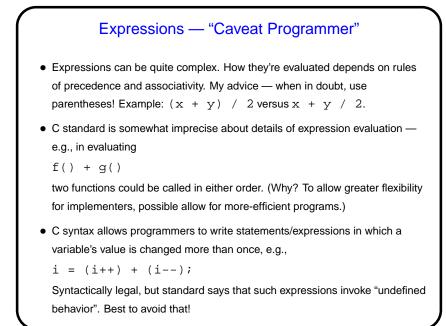


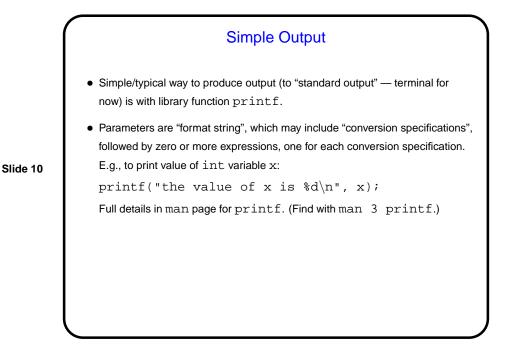




Slide 8

Pre/Post Increment/Decrement
(These four operators are likely new to Scala programmers.)
x++ and ++x both have the side effect of adding 1 to x, but considered as expressions they have different values (before-increment and after-increment respectively). Similarly for x-- and --x.
Often used solely for side effect (e.g., as a substitute for the more-verbose x+=1), but not always (i.e., sometimes used in contexts where expression value matters too).





\int	Simple Input
	 Simple way to get integer/float input (from "standard input") is with library function scanf. Parameters are "format string" (similar to the one for printf) and list of pointers (more later) to variables, e.g.: scanf("%d %d", &var1, &var2); Behaves somewhat like library functions for reading from standard input in other languages, except that it skips whitespace (including newlines) and stops when it encounters something other than what it needs (e.g., non-numeric characters when number is wanted).
	 Considered as an expression, call to scanf has a value, namely the number of variables successfully read. C-idiomatic way to check for success is if (scanf("%d %d"&var1, &var2) == 2)

Sidebar — Man Pages, Revisited

- As mentioned earlier, most commands and many library functions have "man pages" (short for "manual"). These are meant as online references rather than tutorials, so not always easy reading, but usually very complete.
- man program shows its output to you using a program intended for paging through text. On our systems, default is less. Keystroke commands include space to go forward, b to go back, q to quit. h for help — or, of course, you could read all about it (how?).
- Sometimes there are multiple commands/functions with the same name. printf is one. man printf tells you about the (command-line) command, not the C library function. To get all possibilities, man -a printf. To get the one for the library function, man 3 printf.
- Slide 12

```
Minute Essay
• A student sent me the following program and asked why it printed zero for
both x and x+1. What's going wrong? (Hint: gcc -Wall gives warnings.)
#include <stdio.h>
int main(void) {
    int x = 0;
    printf("hello world\n");
    printf("x = %.2f x+1 = %.2f\n", x, x+1);
    return 0;
}
```

