

Slide 1

### Administrivia

- Homework 1 on the Web. Due next week. Meant to be fairly easy starter problems; you can (and probably should) do them using only what we've talked about so far.  
Turn in by e-mail. *Please* mention course name/number and assignment in subject line.

Slide 2

### C Basics — Review/Recap

- 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 3

### Conditional Execution

- Also 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 *statement1* and *statement2* can be single statements or blocks enclosed in curly braces (and should probably be indented, for the convenience of human readers).
- You can build up chains of conditions by making the statement after `else` another `if`, and you can omit the `else` and following statement. (The ideas here should be very familiar; only the syntax should be new.)

Slide 4

### Conditional Expressions

- Scala and Python both provide a way to include if/else idea within an expression.
- C does too, but it's not as obvious — “ternary operator”, e.g.,

```
int sign = (x >= 0) ? 1 : -1
```

## Simple Input Revisited

Slide 5

- 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) ....`

## Example — Finding Roots of a Quadratic Equation

Slide 6

- As an example of all of this, let’s write a program that finds and prints the root(s) of a quadratic equation of the form

$$ax^2 + bx + c = 0$$

using the familiar(?) formula

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- (We’ll also include in this program an example of getting input from standard input.)

### Sidebar — Man Pages, Revisited

Slide 7

- 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`.

### Minute Essay

Slide 8

- Strictly speaking `scanf` reads “from standard input” and `printf` writes “to standard output”. Why say it that way rather than “from the keyboard” and “to the terminal”?

### Minute Essay Answer

- By default “standard input” is the keyboard and “standard output” is the terminal, but I/O redirection allows for other possibilities (from/to files, pipes, etc.).

Slide 9