Slide 1

Administrivia

- Homework 2 is on the Web. Due a week from today.

Slide 2

Minute Essay From Last Lecture

- Yes, some of the reading is easier if you have access to a computer on which you can try things. (If you have a computer connected to the Trinity network, you should be able to access our machines remotely.)
- Yes, there is more in the book than we will have time to cover in class, possibly more than you will need to know to do well in the course. Okay to skim on a first reading and later read more carefully the parts you need (to understand something from class or to do an assignment).

Slide 3

A Few More Scala Basics

- (Last time there was some confusion about calling methods — when you need parenthesese and when you don't. If there are no parameters, it depends on the method. As far as I know, leaving the parentheses out will always work; putting them in will work only with some methods. More later in the semester.)
- The `math` object has many useful methods, including methods to find $\sqrt{n}$, methods for the various trig functions, and constants for π and e.

Slide 4

Example Revisited

- Last time we wrote a simple program to take a number of pennies and "make change" (count it out in dollars, quarters, etc.).
- We wrote two versions, one using a functional style (`val` variables only) and one an imperative style (`var`). Which is better? For now — the one that makes more sense to you.

Slide 5

Conditional Execution

- So far all our programs have executed the same statements every time, just maybe with different numbers.
- Often, though, we want to be able to do different things in different circumstances — for example, print an error message and stop if the input values don't make sense (such as a negative number for the program to make change).
- So, Scala (like most languages) provides some constructs for *conditional execution*. Before we talk about them, we need . . .

Slide 6

Boolean Expressions

- A *Boolean value* is either *true* or *false*; a *Boolean expression* is something that evaluates to true or false.
- We can make simple examples in Scala using familiar math comparison operators (except that the ones for which the keyboard doesn't have a symbol require more than one character). Examples:
  - `x > 10`
  - `y <= 5`
  - `x == y` (**NOTE** the use of == and not =.)

Slide 7

Boolean Expressions, Continued

- *Boolean algebra* defines some operators on these values; the most important for us are written in Scala as
  - `!` — "not", true if the operand is false.
  - `&&` — "and", true if both operands are true.
  - `||` — "or", true if either operand is true (or both are).
- Can use these to build up complex expressions. As with arithmetic expressions, use parentheses when in doubt. Examples:
  - `(x >= 0) && (x <= 10)` (What if we just write `0 <= x <= 10`?)
  - `!(x == y)` (though we could also just write `x != y`).

Slide 8

Boolean Expressions in Scala

- Scala has a type for boolean values (`Boolean`) with the obvious values.
- One thing to know is that the operators for "and" and "or" do not evaluate both operands if the value of the first operand determines the result.

Slide 9

Conditional Execution — `if/else`

- To execute a statement if an expression evaluates to true, use `if`:

```
if (x > 0)
    println("greater than zero")
```

- To execute one statement if an expression is true, another if it's false, use `if` and `else`:

```
if (x > 0)
    println("greater than zero")
else
    println("not greater than zero")
```

Slide 10

`if/else` in Expressions

- Similar rules apply within expressions, e.g.,

```
if (x < 0) -1 else 1
```

has the value -1 if `x` is less than zero, 1 otherwise.

- Many programming languages have a similar construct but express it different; in C and Java the equivalent expression is

```
(x < 0) ? -1 : 1
```

Slide 11

`if/else` and Blocks

- To execute a group ("block") of statements rather than just a single statement, use curly braces for grouping:

```
if (x > 0) {
    println("greater than zero")
    println("and that is good")
}
else {
    println("not greater than zero")
    println("and that is bad")
}
```

Slide 12

`if/else`, Continued

- What happens if you forget the braces? The program may still run, but it probably won't do what you meant.
- Several styles for where to put the curly braces. Which is best? Some people care; I say pick one that's readable and stick with it.

Slide 13

Example — Calculating Grades

- As a simple example, consider a program to calculate numeric and letter grades as described in the syllabus . . .

Slide 14

Minute Essay

- None — quiz.