

Administrivia

- Reminder: Homework 5 due today, Homework 6 Tuesday. (If you can't finish by a deadline, turn in what you have, then a better version later.)

Slide 1

Minute Essay From Last Lecture

- (Almost no one had time to try. One solution is in lecture notes/slides online.)

Slide 2

Loops in Scala — Review/Recap

Slide 3

- Scala provides three loop-like constructs — `while`, `do while`, and `for`.
- `while` and `do while` work as their names might suggest (and the difference is in whether the test to check whether the loop should continue happens before each iteration or after). These constructs would likely be familiar to programmers who know some other imperative language. They're a little un-Scala-like because they require mutable (`var`) variables.
- `for` is actually not a loop but a “comprehension” — a way of repeating some operation on all elements of a sequence. However, in some of its simple forms it would be familiar to programmers who know another imperative language.

Examples, Continued

Slide 4

- Last time we wrote a program to find all primes up through some maximum value. May be our first program that does something you could not more easily do another way?
- Also something you might want to combine with input and/or output redirection, e.g.:

```
echo "1000000" | scala primes.scala > primes.txt
```

and then you could use `vim` to look at `primes.txt` and count how many primes, etc.

Examples, Continued

- As another example, let's finally write that improved "checkbook balance" program that would maintain a list of transactions mentioned a while back.
- (To be finished next time.)

Slide 5

Minute Essay

- None — quiz.

Slide 6