Slide 1

Administrivia

- (None now — I'll ask about office hours as the minute essay question.)

Slide 2

Linked Data Structures — Review/Recap

- Arrays are one example of "collection" data type; they're also an implementation of a general "linear list" type. Another way to implement the same idea is as a "linked list".

  (Review code.)

- Many other types of collections could be defined with similar data structures, for example a "tree" that can represent a hierarchy.

Slide 3

Functions as Parameters to Other Functions

- There are situations in which it's useful to allow functions to use other functions as parameters.
- One example is sorting — the same algorithm applies to sorting anything for which there's a well-defined "less than" operator, so sorting `int`s is much like sorting `double`s, which in turn is much like sorting strings (except that things get a little tricky there because they have varying lengths). So — maybe a general sort function, with one parameter that represents the "less than" operation to use?
- Another example is our numerical-integration program — we could use much the same code to perform numerical integration on different functions if we could somehow make the function to be integrated a parameter.

Slide 4

Functions as Parameters to Other Functions, Continued

- Some languages provide nice built-in support for this idea ("functions are first-class objects"). Examples go back to early "functional languages" and include several more-recent languages such as Scala, Python, and Java (though it's a little clumsy in Java).
- C provides a way to get this effect, via "function pointers".

Slide 5

Function Pointers in C

- The type of a function pointer includes information about the number and types of parameters, plus the return type.
- Example — last parameter to library function `qsort` (in its `man` page). Call this by providing, in your code, a function with declaration

```
int my_compare(const void *, const void *);
```

and using `my_compare` as the last parameter to `qsort`.

(Revised sample program.)

Slide 6

Function Pointers

- Another good use would to something generalizes the code we wrote to approximate π with numerical integration:
- Pretty much the same calculations could be used to approximate any definite integral; just provide start and end points and function.
- (Look at code.)

Slide 7

Homeworks

- (Questions about Homeworks 9 and/or 10?)

Slide 8

Minute Essay

- About office hours during reading days and finals:

  One option is to schedule a couple of hours of "open lab" during reading days (basically, something like the ACM tutoring, but with just me). Times I could do this are between noon and 6pm Wednesday or after 4:30pm Thursday.

  Another option is office hours Monday and Tuesday of the following week, after 3pm (I have exams that finish then).

  Are you interested, and what times are okay for you?