Slide 1

Administrivia

- Reminder: Homework 8 due today.
- Homework 9 on the Web.

  This one may not be easy, but more because you need to understand some parts of the starter code than because you have to write a lot of new code. And I think it's kind of interesting?

Slide 2

2D Arrays Revisited — Conway's Game of Life

- Many years ago mathematician John Conway came up with a simple "game" in which you define a 2D grid of cells, each containing an organism, or not, and rules for updating the grid. (You may have seen this? if not, Wikipedia article seems good. Also Homework 9 describes it and includes a simple text-based version of the game.)
- What would a C program to "play" this game look like?

Slide 3

Conway’s Game of Life

- A program to “play” this game is not so very hard to write:
- The basic calculation is fairly straightforward: Keep two copies of the “board”. At each step, for each cell in the board compute its next value based on its current value and current values of neighboring cells, producing an updated board. Then copy this updated board to the first board and repeat.
- Getting initial values for the board and displaying it can be non-trivial: Really this seems like something where you want a GUI, which is beyond the scope of this course. A simple text-only approach is more doable but still (for the input anyway) somewhat tedious.

Slide 4

Conway’s Game of Life, Continued

- In Homework 9, you get starter code for a simple text-based program to “play” this game.

  The program doesn’t prompt the user for anything; it requires command line arguments that specify an input file and a number of steps. The input file has a text representation of an initial configuration.
- It might be worthwhile to sketch out how I developed this program; we haven’t done a lot of examples in class that would benefit from having several functions.

Slide 5

Program for Game of Life

- Start by trying to break this down into steps:
- Process command-line arguments.
- Read the input file and use it to build a board (2D array) with the initial configuration.
- Build a second board (another 2D array). (Hm, two places where I want to build a 2D array. Make that a function?)
- Print the initial configuration.
- Loop over number of steps, each time updating the board, printing the updated board, and copying the second board to the first.
- Now look briefly at code. Only thing I think is tricky is how I bypass actually copying the updated board.

Slide 6

Homework 9

- Homework 9 asks you to turn in two programs. One is the starter code with the function to update the board filled in.
- The other . . . One of my goals for this assignment was to have you do something that needed a "big" array. Something that seems interesting to me is to experiment with generating initial board configurations "randomly" and then observing how the number of live cells evolves. So you'll write an adaptation of the first program that does that.
- I *think* most code is either straightforward or reasonably well commented.

Slide 7

Minute Essay

- None — quiz.