Administrivia

- Reminder: Homework 7 due today.

- Homework 8 on the Web; due in two weeks.

Slide 1

Homework 6 Essays

- A couple of people mentioned that it was interesting to get some practice working with files in C.

- One person asked about pointer arithmetic on types other than `char`: It works in terms of the size of the type (e.g., if `p` is a pointer to `int`, `p+1` points to the next `int` not the next byte).

Slide 2

- One person said he "spent too much time working inefficiently" (debugging). Sometimes good to recognize when you need to take a break?

- One person asked why closing files is necessary. For input files, may not matter much, except in big programs that use a lot of files you may run into limits on number of open files. For output files, not closing file may lose some data if output is "buffered" (as it usually is).

Multi-Dimensional Arrays in C, Revisited

Slide 3

- Multi-dimensional arrays in C can be ugly — either allocate with fixed size or use VLAs (probably not great for big arrays).

- Now that we know about dynamic memory allocation and more about pointers, can do better. Looking only at 2D arrays for now, two approaches. Both involve representing array as array of arrays/pointers:

- One way is to first allocate array of pointers and then fill it with pointers to dynamically-allocated 1D arrays. (Example code.)

- Another way is to first allocate array of pointers and 1D array big enough to hold whole array, and then fill array of pointers with pointers into this big array. (Example code.)

Homework 8 — Conway's Game of Life

Slide 4

- 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.)

- A program to "play" this game is not so very hard to write (continued on next slide).

Conway's Game of Life, Continued

Slide 5

- 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.

Conway's Game of Life, Continued

Slide 6

- In Homework 8, you get starter code for a simple text-based program to "play" this game. As with the past few homeworks, 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 many functions.

  (I first assigned this problem in my CSCI 1312 class (CS1 for ENGR), and one of my goals for it was to have students do something that needed a "big" array. But I think it's an interesting problem for this class too! The next slide is targeted more toward CS1-level students.)

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.

Slide 7

- 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.

Homework 8

- Homework 8 asks you to turn in two programs. One is the starter code with the function to update the board filled in.

- The other . . . 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

Slide 8

that does that.

- I *think* most code is either straightforward or reasonably well commented.

Minute Essay

- Had you seen Conway's Game of Life before?

Slide 9