Arrays as Pointers

10/26/2007

Opening Discussion

- Let's look at solutions to the interclass problem.
- Have I ever run into a problem that I couldn't solve with a C program?
- Counting lower case letters in a string.
- Do you have any questions about the assignment?

Arrays as Pointers

- As I mentioned last time in class, arrays are basically pointers that are set to point to a chunk of memory of the proper size.
- The differences are that you can't change what they point to and the memory for them to point to is set up for you.
- We can verify this by printing the address of an array as well as the array.
- We can also write functions that take pointers and pass in arrays.
- a[i] is the same as *(a+1).

Using Pointers to Walk Arrays

- All the loops that we have done where we walk through arrays could have been done with pointers and pointer arithmetic.
- Let's write pointer versions of the copy and length string functions.
- There was a time when these were more efficient than the array versions. Compilers have improved so that this is generally not true and most people find the array versions easier to read.

Multidimensional Array as Pointer

- If an array is just a pointer to a chunk of reserved memory, what is a multidimensional array?
- Your book has a description that I want to test, because I've read other things that make me question.
- We'll run a test to see what gcc does.

Passing Arrays to Functions

- Because of the similarity of pointers and arrays, any function you want to pass a 1-D array into, you can pass a pointer instead.
- Because of compiler optimizations and the fact that the array syntax is more clear, you shouldn't have much need to go to the pointer syntax, but you can.
- Passing multidimensional arrays into functions is a different story. Here you have to provide the size of one or more indexes and you can't do that with pointers.

Minute Essay

- What are your plans for the weekend?
- Interclass Problem – Benchmark two versions of a function that deals with an array or a string. Use array notation for one and pointer notation for the other. You'll have to call it many times for this to work. Use the Linux command “time” to see how long they take to run. You want run times of at least a few seconds.