

Recursion: Part 3

10-30-2002

Opening Discussion

- What did we talk about last class?
- Do you have any questions about assignment #6?
- Practice with recursion.

Recursion in Multiple "Directions"

- Last time we looked at recursive functions that call themselves more than once.
- These functions can do things that aren't easily done with loops.
- What really happens is that it completes one of the calls completely, then makes the others. The memory of the stack is what allows it to come back and go in other directions.

Divide and Conquer

- Another example of where recursion can be very useful is in algorithms that are called divide and conquer algorithms.
- In these algorithms, the problem is repeatedly broken into smaller pieces until it gets to something that can be easily dealt with.
- It then combines the small pieces to get the final answer.

Simple Examples

- Let's write code for doing some simple problems through divide and conquer. These problems display the technique, not so much their power.
 - Sum the elements of an array.
 - Find the minimum element in an array.

Maze Example

- Along the lines of the flood fill or maze problems, let's write a function that tells us how many ways there are to get from a start point to a finish point in a given maze.

NULL Pointers

- In our earlier discussion of pointers we looked at how to declare pointer variables and assign them the address of other variables.
- Sometimes we also want a value that we can use when a pointer doesn't point to a valid address instead of just garbage.
- NULL is this value. It is defined in `stdlib.h`.

Minute Essay

- Write code that uses divide and conquer to count the number of even numbers in an array of integers.
- I won't be here Friday, but you will have a lecture on file access. Remember that you should be working on assignment #6.
