Assignment 5
Due: Thursday 19 February 2004  10:00 A.M. EST


2. ForTran, COBOL, and BASIC had no pointers in the language. However, they were Turing-complete languages. Explain how someone could implement a complex, dynamic data structure in a language without pointers. By complex, I mean something with nodes such as a linked list or a binary tree or something. By dynamic, I mean nodes can be added or removed at any time. It has to be a good implementation, too. Your pointer-free binary tree, for example, has to run roughly as fast as a regular binary tree would. If there are some drawbacks to your implementation, be sure to list them. Your strategy should be general enough to be adaptable to any complex, dynamic data structure.

3. Design a limited but effective way to handle dynamic allocation on the heap. Assume all allocations will be for the exact same amount of memory. Heap memory can thus be divided into $k$-byte chunks. When someone requests a chunk, you have to give him a chunk, but not a chunk already in use. When someone releases a chunk, that chunk has to be available for someone else down the road. And whatever data structure you use to keep track of available and unavailable memory has to be stored on the heap, too. You don't have a whole extra piece of memory to keep track of this information. This strategy should work no matter how large or how small your heap is.

4. There are three kinds of loops commonly used in programming languages: pre-condition, post-condition, and mid-condition loops. Pre-condition loops test the condition before running the loop: while (test) { A }. Post-condition loops test the condition after running the loop: do { a } while (test). Mid-condition loops (which do not exist in C/C++/Java) put the test in the middle of the loop: do { a } while (test) { b }. Such a loop would perform the a stuff exactly one more time than it would perform the b stuff. Show that these three loops are equally powerful by showing how to use each kind to emulate the other two. You are not allowed to use break, continue, or any other flow-of-control statement in your emulation, but you are allowed to use if statements.