Final Exam Study Guide

The exam is closed book and closed notes. It is comprehensive and includes all materials covered during the term so far, in lecture, lab, and on the projects.

Topics covered:

- ADTs: string, set, stacks, queues, lists, container/iterator, binary search trees, n-ary trees and any ADTs used in projects
- All algorithms used in ADTs, projects, or covered in lecture
- Pre/Post/In fix notations, conversion algorithms
- Classes, operator overloading
- Copy semantics: constructors, destructors, assignment, constant time swap, copy, this, copy semantics
- Templates
- Recursion
- Dynamic memory allocation (new, delete), dynamic arrays, linked lists, double linked lists, trees, pointer arithmetic
- Tree traversal algorithms – inorder, preorder, postorder
- Simple inheritance and virtual methods
- Software testing and constructing test cases
- Documentation – pre and post conditions, class invariants

Not on the exam: make, svn, or Unix commands

Sample Exam Questions

These questions are representative of the types and format of the exam questions. The instructor has given similar questions in the past.

1. What are the three components of an abstract data type (ADT)?

2. Write a template class for the ADT Stack. You must use a linked list. You must implement the following:
   - constructor
   - destructor
   - copy constructor
   - assignment
   - constant time swap

3. Write a template class for the ADT binary tree using dynamic memory. Implement copy semantics. The copy constructor and destructor must be recursive.

4. Convert the following expression to prefix and postfix: A + B + C * D * E