CS 235 Final Exam

Instructor: R. P. Burton

June 18-19, 2014 (Wednesday and Thursday)

Due in the Lab on Thursday no later than 8:00 p.m.

Penalty for submitting the final late:

No Credit

Open Book (142 course text and your CS 235 course text only), Open Notes (including your own Lab solutions)

Open Secondary Storage Device: yours only

Open Laptop: if you wish

Open Course Website and www.cplusplus.com, but no other Internet resources (no Google)

Closed Neighbor (and everyone is thy neighbor)

**\*Instructions\***

(Please read carefully)

1. This test consists of a C++ programming problem. Read and understand the statement of the problem completely before beginning to design, code, and test. As part of your design, consider the test cases that will establish the correctness of your solution. Test your solution thoroughly before submitting it.
2. Produce a solution, which consists of your C++ code, with a comment at the beginning of each file (both .h and.cpp) which includes your name and “CS 235 Spring 2014 Final Exam.” Upload your completed project by compressing the files and submitting them through Learning Suite with TA assistance. If a packet is not collected by a TA upon submission, you will not be graded and will therefore receive no credit for the exam (ouch!). Your solution packet must all be stapled together before it will be accepted by a TA. No time stamps will be accepted for this exam. Incorrect files submitted for any reason will result in a 0 on this final exam.
3. Attribute any code taken from or based on other sources (excluding the course texts and the course websites). Attributed code copied from or based heavily on outside sources is worth half credit. Unattributed code copied from or based heavily on outside sources is worth no credit.
4. Understanding the problem correctly is part of the examination. If something seems unclear, ask a CS 235 TA (but no one else) for clarification. You may pose questions to the CS 235 TAs via the Help Queue. However, the TAs generally are not permitted to answer questions related to design, C++ implementation, debugging, or testing.
5. Prior to submitting your final, score it using the attached scoring sheet (this will help you maximize your points and will help us grade your exam accurately). You will receive 3 points extra credit if your self-grading score is within 5 points of the TA Total score. You will lose 3 points if your self-graded score is more than 10 points different from the TA Total score.
6. You must receive at least 75% on the final in order to receive an A grade in the course.
7. Sign the scoring sheet (1) to certify that you have reviewed your posted scores for the term and alerted a CS 235 TA in writing to any discrepancies, (2) to request that your final exam be graded, and (3) to certify that no unfair information related to this exam has been received by you, either directly or indirectly, and that none will be conveyed by you. If we discover that you cheated or assisted someone in cheating, intentionally or unintentionally (including accidentally), your score for this exam may (and probably will) be rand() % 1.

We’re serious.

**AVL**

In computer science, an AVL tree is a self-balancing binary search tree (BST), and it was among the first such data structure to be published. In an AVL tree, the heights of the two subtrees of any node differ by at most one. Lookup, insertion, and deletion all take O(log n) time in both the average and worst cases, where n is the number of nodes in the tree. Insertions and deletions may require the tree to be rebalanced by one or more rotations.

While an AVL tree is a BST, an AVL tree balances itself so that the right subtree does not critically outweigh the left subtree, and vice versa. For every node in the tree, the height of its subtrees differs by no more than 1. Adding to and removing from an AVL tree can result in an unbalanced tree. There are 4 kinds of rebalancing you must implement, and these are done with rotations. Note that solving an imbalance can result in another imbalance further up the tree. For examples of the 4 kinds of imbalances and how to solve them, refer to the PowerPoint slides.

For the exam you will be given the Tree Display interface, and a test driver referred to as “Convergence”. These things are all found in the compressed file for the exam on LearningSuite.

You do NOT need to pass Valgrind to receive full credit. You will be graded based on how many of the tests you pass on the test driver.

There are several methods for rebalancing; each method may result in a distinct tree. Specific conventions need to be used to ensure that your tree properly matches ours. When rebalancing, refer to the PowerPoint slides for more detailed information on the convention for purposes of this exam.

You should remove nodes from the AVL tree in the same manner used in BST.

Remember to disallow duplicate entries and handle the cases where the element to be removed is not in the tree.

The Tree Display interface is available for the exam. Review the Lab 6 requirement notes for more details on how to use the interface. Also note that the default setting for the Tree Display is a BST. If you wish to test AVL trees, you must either go to File->New->AVL or load an Error Report by going to File->Open->Error Report.

**Scoring Sheet**

Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ TA Initials\_\_\_\_

**Student:** **TA: Tests:**

\_\_ / 5pts \_\_ / 5ptsAdd “5”.

\_\_ / 5pts \_\_ / 5pts Remove “5”.

\_\_ / 5pts \_\_ / 5pts Remove “0”.

\_\_ / 10pts \_\_ / 10pts Add 0 – 9, then add duplicates.

\_\_ / 10pts \_\_ / 10pts Remove 0 – 9, then remove 0 – 9 again.

\_\_ / 10pts \_\_ / 10pts Add 500 – 0 in increments of 15, then add several duplicates.

\_\_ / 15pts \_\_ / 15pts Remove 0 – 500 in increments of 5, then remove the same integers again.

\_\_ / 5pts \_\_ / 5pts Empty list.

Convergence test:

\_\_ / 15pts \_\_ / 15pts Add 0 – 50, 0 - -50. (1 of 4)

\_\_ / 15pts \_\_ / 15pts Remove 0 – 50, 0 - -50. (2 of 4)

\_\_ / 15pts \_\_ / 15pts Add 0 – 50, 0 - -50. (3 of 4)

\_\_ / 15pts \_\_ / 15pts Remove 50 – 0, -50 – 0. (4 of 4)

\_\_ / 5pts \_\_ / 5pts Empty list.

\_\_ / 20pts \_\_ / 20pts Convergence test with varying increments.

\_\_ / 150 \_\_ /150 Subtotal

\_\_ / Passes Valgrind (10 Points)

\_\_ / Accurate Scoring Bonus(+3/-3)

\_\_\_ / **Total**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student Signature TA Grader