# CS 2150 Final Exam

## Name _____

You MUST write your e-mail ID on EACH page and bubble in your userid at the bottom of EACH test page, including this page. And put your name on the top of this page, too. You don't have to bubble in the reference pages, as they are not going to be scanned in.

If you are still writing when "pens down" is called, your exam will be ripped up and not graded – even if you are still writing to fill in the bubble forms. So please do that first. Sorry to have to be strict on this.

Other than bubbling in your userid at the bottom, please do not write in the footer section of each page.

There are 10 pages to this exam. Once the exam starts, please make sure you have all the pages. Each page is worth 12 points, for a total of 108 points on this exam.

**If you do not bubble in a page, you will not receive credit for that page!**

This exam is CLOSED text book, closed-notes, closed-calculator, closed-cell phone, closed-computer, closed-neighbor, etc. Questions are worth different amounts, so be sure to look over all the questions and plan your time accordingly. Please sign the honor pledge below.

_____

_____

_____

_____

_____

*Three things are certain:*
*Death, taxes, and lost data.*
*Guess which has occurred.*

(the bubble footer is automatically inserted into this space)

## Page 2: C++

1. [3 points] Give two compelling uses of unions.

2. [3 points] Explain the difference between static dispatch and dynamic dispatch. How do you make C++ use one over the other?

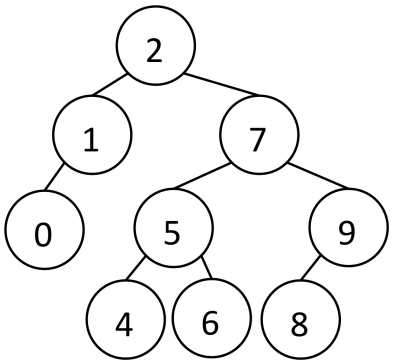
3. [3 points] Other than syntax, what are the three main differences between references and pointers?

4. [3 points] The run-time error you have probably run into the most this semester is "segmentation fault". What does this error name mean and why is it called that?

(the bubble footer is automatically inserted into this space)

## Page 3: Trees and Lists

5. [4 points] Insert 3 into the AVL tree below. Show the resulting data structure.




6. [8 points] Consider the four types of binary trees that we have seen in the course: BST, AVL, splay, and red-black. Give one advantage and one disadvantage of each type of tree.

## Page 4: Hash Tables

7. [6 points] Consider the four types of collision resolution strategies that we have seen in this course: separate chaining, linear probing, quadratic probing, and double hashing. Give one advantage and one disadvantage of each type of collision resolution strategy.

8. [3 points] Assuming a probing collision resolution strategy, what is the running time of a hash table's `insert()`, `find()`, and `remove()`? *Briefly* explain why in each case.

9. [3 points] Why should hash table sizes always be prime? Give two reasons, and *briefly* explain each one.

(the bubble footer is automatically inserted into this space)

## Page 5: x86

10. [3 points] What are the steps necessary in the x86 C calling convention caller's prologue?

11. [3 points] What are the steps necessary in the x86 C calling convention caller's epilogue?

12. [3 points] What are the steps necessary in the x86 C calling convention callee's prologue?

13. [3 points] What are the steps necessary in the x86 C calling convention callee's epilogue?

(the bubble footer is automatically inserted into this space)

## Page 6: Heaps

14. [12 points] Consider the merge operation on heaps: given two separate heaps (each of, say, $n$ elements), their elements are merged to create a single heap with $2n$ elements. This is easily achieveable in $\Theta(n \log n)$ time – take out the $n$ elements of the second heap, and insert each one (in $\log n$ time) into the first heap. However, that's too slow for our application. Your task is to design a heap-like data structure that allows for a constant-time merge operation. You must still have the other heap operations (`insert()`, `deleteMin()`, `findMin()`), and they must be as efficient as possible.

    a. [6 points] Describe, at a high level, how your data strcture will work. Assume that we know all about heaps, so just explain the differences with that.

    b. [6 points] Will this modify the running time of the other heap operations (`insert()`, `deleteMin()`, `findMin()`)? Explain why or why not. And if it will change their running times, what will the new times be?

## Page 7: Huffman Coding

15. [12 points] Consider the letter frequencies shown below – this is for "to be or not to be", a line from Shakespeare's Hamlet.

    a. [8 points] Construct a Huffman coding tree based on this data. You will need to show your intermediate steps to receive any partial credit.

| Ltr | Freq |
|---|---|
| b | 2 |
| e | 2 |
| n | 1 |
| o | 4 |
| r | 1 |
| t | 3 |

    b. [4 points] Based on your Huffman coding tree from part (a), determine the prefix codes. Note that if you make *some* errors in your Huffman tree, you can still get credit for this part; but if your tree is totally wrong, you can't.

## Page 8: Graphs

16. [6 points] Write the algorithm – in pseudo-code – for Dijkstra's shortest path.

17. [6 points] *Briefly* describe the difference between how Prim's and Kruskal's minimum spanning tree algorithms work. You need not remember which is which, so if you mix up the names, that's fine.

## Page 9: Miscellaneous

18. [3 points] Name three unique featuers of INTERCAL.

19. [3 points] List eight Emacs keyboard shortcut commands.

20. [3 points] List eight GDB commands.

21. [3 points] What are the two properties of memory accesses that allow caches to work, and briefly what do they mean?

(the bubble footer is automatically inserted into this space)

## Page 10: Demographics

We meant to ask these in an end-of-the-semester survey, but didnt get to it in time. So we'll put it here for some extra points on the exam!

22. [0 points] Did you put your name and userid at the top of this page? You need to in order to get the points on this page.

23. [2 points] What is your major or minor (if you have not declared, then your intended major or minor)?

    - BS CS
    - BA CS
    - BS CpE
    - CS minor
    - Neither majoring nor minoring in computing
    - Other (please explain): _____

24. [2 points] What CS 1 class did you take? Please circle one.

    - CS 101 (now 1110)
    - CS 101-E (now 1111)
    - CS 101-X (now 1112)
    - CS 150 (now 1120)
    - AP credit
    - Transfer credit
    - Placed out of it via the 101/1110 placement exam
    - Other (please explain): _____

25. [2 points] If you took your CS 1 class in college (i.e. CS 101/1110, 101-E/1111, 101-X/1112, 150/1120, or a transfer class), in what semester did you take it?

26. [2 points] What CS 2 class did you take? Please circle one.

    - CS 201 (now 2110)
    - CS 205 (now 2220)
    - AP credit
    - Transfer credit
    - Other (please explain): _____

27. [2 points] If you took your CS 2 class in college (i.e. CS 201/2110, 205/2220, or a transfer class), in what semester did you take it?

28. [2 points] Did you attend the review session? You'll get full credit for this question, as long as you answer it honestly (we know most of the people that were there, but not all).

(the bubble footer is automatically inserted into this space)