

# CSCI 204 Study Guide for Exam Two

Prof. Markstrum

Fall 2009

## 1 The Rules

- You may bring one 8x11 information sheet of paper with anything you like **hand-written** on one side. No printed information is allowed.
- Any code or information which comes from the Internet, a non-cs204-course textbook, or another person needs a citation. Information from the Internet or from an outside book or person may only be used on this exam with cites and at the instructor's discretion. Always ask if you are unsure.
- Do **not** bring any electronic devices such as calculator, PDA, cell phone, or iPhone. If you have to have one with you (such as a cell phone), turn it off and don't take it out.
- You must staple your information sheet to the exam before you hand it in. (I provide the stapler). I reserve the right not to grade your exam if you forget to include your information sheet. If you don't use one, please indicate so on the cover sheet of the exam.
- The exam will begin at 11:00 AM promptly and will end at 11:52 AM promptly. If I make an exam too long, I will deal with it later.
- Do not discuss the exam with anyone else except the instructor until after the exam is over and you are certain they already took it.
- Bring several writing implements and an eraser.

## 2 What's on the exam?

This exam covers Exceptions thru (including) Lists. The format may include multiple choice questions, short answer questions, and code writing questions. At this stage, I expect you to answer any questions about Lists (including implementation questions).

## 3 A partial list of subjects

This list may not be complete. You should be able to

- Understand Java generics (including bounded type parameters) and exceptions.
- Sort numbers using bubble sort, selection sort, insertion sort, merge sort or quicksort. You should know the basic idea well enough to do this even if code is not provided.
- Do basic complexity analysis (big-O notation) for assignment statements, loops (for-loop, while-loop), selections (if), and simple recursions.
- Understand how to compare big-O expressions.

- List and explain the standard list operations or write a decent javadoc comment for them (the specification).
- Draw memory pictures for linked lists (including doubly linked lists and a circular doubly linked lists).
- Use appropriate comments in your code
- Analyze the run time of list operations or recursive algorithms (including sorting algorithms).