

CMPSC 112
Introduction to Computer Science II
Fall 2016

Exam 2 Study Guide
Delivered: Tuesday, December 13, 2016
Exam 3: Thursday, December 15, 2016, 7:00 pm

Introduction

This course will have its second exam on Thursday, December 15, 2016 from 7:00 to 10:00 pm. The exam will be “closed notes” and “closed book” and it will cover the following materials. Please review the “Course Schedule” on the web site for the course to see the content and slides that we have covered to this date. Students may post questions about this material to our Slack team. The questions on the examination will be drawn from the content in *Data Structures and Algorithms in Java* (DSAAJ) by Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser.

- All prior sections of the textbook covered for the first examination
- Chapter One in DSAAJ, all sections (i.e., “Java Primer”)
- Chapter Two in DSAAJ, all sections (i.e., “Object-Oriented Design”)
- Chapter Three in DSAAJ, only Section 3.1 (i.e., “Fundamental Data Structures”)
- Chapter Three in DSAAJ, only Sections 3.2–3.5 (i.e., “Linked Lists”)
- Chapter Four in DSAAJ, skipping Section 4.4 (i.e., “Algorithm Analysis”)
- Chapter Five in DSAAJ, skipping Section 5.6 (i.e., “Recursion”)
- Chapter Six in DSAAJ, all sections of the chapter (i.e., “Stacks, Queues, and Deques”)
- Chapter Seven in DSAAJ, skipping Sections 7.6 and 7.7 (i.e., “Lists and Iterators”)
- Chapter Ten in DSAAJ, only Sections 10.1–10.2 (i.e., “Maps, Hash Tables, and Skip Lists”)
- Using the many commands in the Linux operating system; editing in `gvim`, compiling and executing programs in Linux; knowledge of the basic commands for using `git` and Bitbucket.
- Your class notes and lecture slides and all of the laboratory assignments.

The examination will include a mix of questions that will require you to draw and/or comment on a diagram, write a short answer, explain and/or write a source code segment, or give and comment on a list of concepts or points. The emphasis will be on the following list of illustrative subjects. Please note that this list is not exhaustive — rather it is designed to suggest representative topics.

- Key programming constructs in Java (e.g., variables, conditional logic, and iteration).
- How to call and implement a method in the Java programming language.
- The purpose of `import` statements and the `CLASSPATH` in the Java language.
- Declaring and using arrays and random number generators in the Java language.
- Object-oriented design concepts (e.g., inheritance, encapsulation, and exceptions).

- The tools and concepts associated with the engineering of software (e.g., build systems).
- How to document the source code of a method using the JavaDoc commenting standard.
- Experimental and analytical evaluation of algorithms (e.g., using timers and Big-Oh notation).
- The use of the doubling method to understand the worst-case performance of an algorithm.
- The steps for performing an asymptotic analysis of an algorithm's time complexity.
- The use (and misuse) of recursion in Java programs that repeatedly perform an action.
- The structure of a singly and doubly linked list and the nodes of which they are comprised.
- The strengths and weaknesses associated with using singly, doubly, and circularly linked lists.
- The trade-offs associated with using either arrays or linked lists to implement data structures.
- The implementation trade-offs of the stack abstract data type with an array or a linked list.
- The applications of the stack data type (e.g., reversing an array or matching HTML tags).
- The implementation trade-offs of the queue abstract data type with an array or a linked list.
- The applications of the queue data type (e.g., process scheduling on an operating system).
- How to design and implement array lists and dynamic arrays in an object-oriented language.
- The various features provided by the array list and vector in the `java.util` package.
- The benefits associated with storing data inside of an instance of the tree abstract data type.
- The definitions and properties of the tree (i.e., both concrete examples and formal definitions).
- The meaning of the term “depth of a node” and “height of the tree” in the context of trees.
- The similarities and differences between the structure of general and binary trees.
- An intuition about how the hashtable stores key-value pairs in an associative array.
- The various features provided by the maps and hashtables in the `java.util` package.
- The applications of the hashtable abstract data type (e.g., counting word frequencies).
- The benefits and challenges associated with storing data inside of a hashtable.
- The worst-case time complexities of all of the data structures studied during this course.
- Practical laboratory techniques (e.g., editing, compiling, and running programs; effectively using files and directories; correctly using Bitbucket through the command-line `git` program).
- Understanding Java programs (e.g., given a short, perhaps even one line, source code segment written in Java, understand what it does and be able to precisely describe its output).

Examination Policies

Minimal partial credit may be awarded for the questions that require a student to write a short answer. You are strongly encouraged to write short, precise, and correct responses to all of the questions. When you are taking the examination, you should do so as a “point maximizer” who first responds to the questions that you are most likely to answer correctly for full points. Please keep the time limitation in mind as you are absolutely required to submit the examination at the

end of the period unless you have written permission for extra time from a member of the Learning Commons. Students who do not submit their examination on time will have their overall point total reduced. Please see the course instructor if you have questions about any of these policies.

Reminder Concerning the Honor Code

Students are required to fully adhere to the Honor Code during the completion of this exam. More details about the Allegheny College Honor Code are provided on the syllabus. Students are strongly encouraged to carefully review the full statement of the Honor Code before taking this exam.

The following provides you with a review of Honor Code statement from the course syllabus:

The Academic Honor Program that governs the entire academic program at Allegheny College is described in the Allegheny Academic Bulletin. The Honor Program applies to all work that is submitted for academic credit or to meet non-credit requirements for graduation at Allegheny College. This includes all work assigned for this class (e.g., examinations, laboratory assignments, and the final project). All students who have enrolled in the College will work under the Honor Program. Each student who has matriculated at the College has acknowledged the following pledge:

I hereby recognize and pledge to fulfill my responsibilities, as defined in the Honor Code, and to maintain the integrity of both myself and the College community as a whole.

Students who have questions about Allegheny College's Honor Code and how it applies to the completion of a quiz or an examination in Computer Science 112, should immediately schedule a meeting with the course instructor to openly discuss their questions and concerns.

Strategies for Studying

As you study for this examination, you are encouraged to form study groups with individuals who were previously, during a laboratory session, a member of one of your software development and empirical study teams. You can collaborate with these individuals to ensure that you understand all of the key concepts mentioned on this study guide. Additionally, students are encouraged to create a Slack channel that can host questions and answers that arise as you continue to study for the test. Even though the course instructor will try to, whenever possible, answer review questions that students post in this channel, you are strongly encouraged to answer the questions posted by your colleagues as this will also help you to ensure that you fully understand the material.

When studying for the test, don't forget that the Web site for our course contains mobile-ready slides that will provide you with an overview of the key concepts that we discussed in the first modules. You can use the color scheme in the slides to notice points where we, for instance, completed an in-class activity, discussed a key point, or made reference to additional details available in the DSAAJ textbook. Students who have additional questions about the content on this review sheet can ask them during our in-class review session. Finally, please remember that while the test is taken individually, your review for it can be done collaboratively!