

Data Structures and Algorithms



Trinity College
Library
Univ. of Dublin

William D McQuain

Email: `wmcquain@cs.vt.edu`

Office: 634 McBryde Hall

Office Hours: see course website

CS 3114 Data Structures and Algorithms

Advanced data structures and analysis of data structure and algorithm performance. Sorting, searching, hashing, and advanced tree structures and algorithms. File system organization and access methods.

Course projects require advanced problem-solving, design, and implementation skills.

Having successfully completed this course, the student will be able to:

- Choose the data structures that effectively model the information in a problem.
- Judge efficiency trade-offs among alternative data structure implementations or combinations.
- Apply algorithm analysis techniques to evaluate the performance of an algorithm and to compare data structures.
- Implement and know when to apply standard algorithms for searching and sorting.
- Recognize and apply design patterns, and make judgments about when a particular pattern will improve a design.
- Design, implement, test, and debug programs using a variety of data structures including buffer pools, hash tables, binary and general tree structures, search trees, tries, heaps, graphs, and B-trees.
- Select appropriate methods for organizing data files and implement file-based data structures.
- Apply object-oriented design principles to data structures in medium-scale software systems.
- Apply design guidelines to evaluate alternative software designs.

CS 2104	Intro to Problem Solving in CS (for CS majors)
CS 2114	Software Design and Data Structures
CS 2505	Computer Organization I
Math 2534	Discrete Mathematics
or	
Math 3034	Introduction to Proofs

All students must have completed each of the CS prerequisites with a grade of C or higher (C- is not acceptable).

There will be absolutely NO exceptions to these requirements.

Note Lying about whether you meet the prerequisites is an Honor Code offense. Students who do so may be charged, at the discretion of the Department of Computer Science.

Matthew Bock

mattb93 AT vt DOT edu

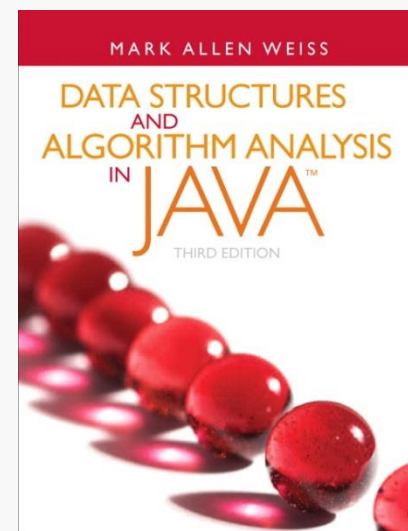
Office hours on course website

Required:

Data Structures and Algorithm Analysis in Java, 3rd Edition

Mark Allen Weiss, Addison-Wesley, ©2012

ISBN 978-0-13-257627-7



Recommended:

CS 3114 Course Notes, Spring 2015 Edition

W D McQuain, ©2001-2015

(available ONLY at the course website)

courses.cs.vt.edu/~cs3114/Spring15/

Final grades will be based on the average achieved over the following :



Item	Weight	Dates
Programming Projects	40%	See website
Homework	20%	See website
Midterm Test*	15%	June 10
Final Exam**	25%	July 1 13:00 – 15:00

** Exam score will replace the midterm score, if it is higher.

Grade Scale

The usual 10-point scale will apply (subject to any curve). A final average of 90% will guarantee an A-, 80% will guarantee a B-, and so forth.

Curve

A grade curve may or may not be employed in this course. The application of a curve is dependent upon class performance on tests, projects and homework. The decision to utilize a curve rests entirely with the course instructor.

Statute of Limitations

Any questions about the grading of an assignment must be raised with your instructor within one week after the graded assignment has been made available to you.

Late Penalties for Projects

Project solutions can be turned in after the posted deadline, in which case a per diem penalty will be assessed.

The penalty will be 10% per day.

Extensions

Any request for an extension must be made, preferably by email, at least 24 hours prior to the due date. Reasons must be valid (generally events beyond your control) and documented.

Late submissions will not be given any credit if submitted after graded assignments or solutions have been released.

Homework

Usually, no late submissions will be allowed for homework assignments.

Evaluation of Correctness, Design and Implementation, and Documentation

The correctness of operation of your programming projects will be evaluated by executing your solution with test data constructed by the course staff. While some test data will be provided, there is no guarantee that data will cover all cases.

Each project will have certain explicit requirements for design and implementation. Your solution will also be evaluated for adherence to those requirements.

You are expected to enter CS 3114 with considerable understanding of good software engineering practice, and you are expected to apply those lessons here. That means that you may be penalized for failing to make good decisions, even if there are no explicit guidelines in the specification of a project.

You are also expected to incorporate professional internal documentation into your projects. See the Programming Standards page on the course website for some suggestions and samples.

Test Environments

- All programming assignments submitted are required to compile with `javac` version 8 update 25 or later.
- Unless specified otherwise, programs will only be tested under that environment.
- It is the **YOUR** responsibility to ensure that **YOUR** programs execute correctly in the appropriate environment; programs that do not will receive substantial deductions.



Students developing with a different Java major version are advised that will almost certainly lead to problems.



Your programs **WILL** be tested with the environment listed above. If it fails to compile, or exhibits incorrect behavior, we don't care that it may compile elsewhere, or appear to run correctly elsewhere.



Why use the command line?

- All programming assignments submitted are required to compile with `javac` version 8 update 70 or later.
- The Eclipse IDE uses its own internal Java compiler; that may differ from the "real" Java compiler from Oracle.
- We will compile from the command line with the "real" Java compiler when we test your projects.
- It is the YOUR responsibility to ensure that YOUR programs execute correctly in the appropriate environment; programs that do not will receive substantial deductions.

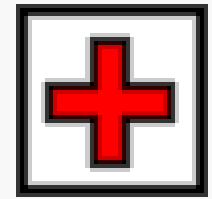


How can I use the command line?

- You have considerable experience with the Linux command line from CS 2505.
- The basic principles are the same for the Windows command line.
- We will post notes on installing the JDK from Oracle and using it from the Windows command line. Things would be very similar on Linux.

General Issues

- CS 3114 classmates
- CS 3114 Forum board
- CS 3114 TA
- CS 3114 Instructor



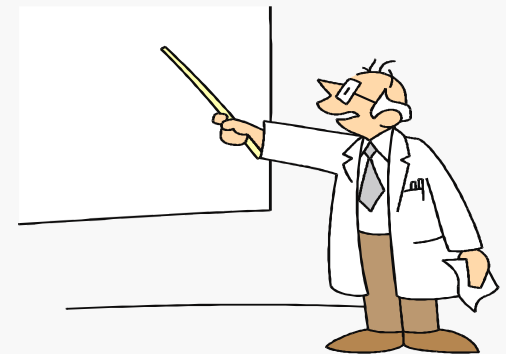
Java Language Help

- CS 3114 Forum
- texts from earlier courses



Lecture Instruction

Lectures will consist of presentations, applications, problems and solutions interspersed with classroom discussion.



Backups

Students are responsible for making backup copies of all their work in this (and all) courses.

Loss of work due to hard drive failure is **NOT** an acceptable excuse. Backup copies of files on the same hard drive are not backup copies. Backup copies of files on second hard drives are also risky. Backup copies should be maintained on two separate distinct storage mediums, (e.g., hard drives and Zip disks).

Backup copies should be maintained until after the end of the term and students have received their course grade. (The Army lives by triplicate for a reason.)

Remember: Computer systems are mechanical devices.

Systems fail. Drives die. Bad sectors appear.

Network connections break.

Plan for it. It is inevitable!



An exhaustive list of Honor Code violations would be impossible to present here, but among other things, each of the following is a flagrant violation of the Virginia Tech Honor Code, and violations will be dealt with severely (Honor Court):

- Working with another student to derive a common program unless group work is explicitly allowed on that assignment.
- Discussing the details required to solve a programming assignment. You may not share solutions.
- Copying source code (programs) in whole or in part from someone else.
- Copying files from another student's disk even though they might be unprotected.

It is acceptable to discuss with classmates a programming assignment in a general way, i.e., to discuss the nature of the assignment. In other words, you may discuss with your classmates what your program is required to accomplish but not how to achieve that goal using Java. In no way should the individual statements of a program or the steps leading to the solution of the problem be discussed with or shown to anyone except those people cited in the following statement.

Feel free to discuss the homework assignments and your program source code with the teaching assistants assigned to CS 3114, the instructor, or the free tutors provided by UPE. The discussion of your program source code must be limited to these people. Note that this specifically excludes discussions of your program source code with other students (even if they are not enrolled in CS 3114), or with tutors except for those named above. Privately hired tutors are not an exception to this requirement, nor are athletic or other tutors provided by the University.

Copies of all submitted work are retained indefinitely by the Department. Submitted programs are subjected to automated analysis for detection of cheating.

If you have any question as to how the Honor Code applies to this class, remember that:

- Any work done in this class must be done on an individual basis.
- Credit will be given only for work done entirely on an individual basis.
- Do not make any assumptions as to who can provide help on a programming assignment.
- All submitted work is archived. All submitted programs will be subjected to automated cheat analysis.

Evidence indicating the violation of the policies stated above will be submitted to the Honor Court.

It is much easier to explain a poor grade to parents or a potential employer than to explain an Honor Court conviction.

In recent terms we have observed a new behavior regarding cheating.

Some of the projects we use each semester may have been used in previous offerings of the course.

A small number of students have submitted solutions that were based (in whole or in part) on solutions submitted by other students in previous offerings. Be advised:

- That is cheating.
- When we do the cheating analysis for a project, we include submissions from previous offerings as well as the current term. (We save everything.)
- If we detect this form of cheating, the students from the previous terms will also be charged with violating the Honor Code.
- The University does make provisions for cases in which a charged student has already graduated. You do not want to discover how this works, so safeguard your code for the long term!

At some point you will very likely want to make samples of the code you write available to potential employers. There is absolutely nothing wrong with that!

But if you do that, you need to be sure you're not making it easy for other students to access your code.

GitHub is an interesting service, but you need to use the protection options that the free student accounts offer.

And there are alternatives...