

# CS 220 – Data Structures

Fall 2011

Mondays and Wednesdays 2:55 p.m. – 4:10 p.m. Dana 318

**Instructor:** Ingrid Russell

**URL:** [uhaweb.hartford.edu/irussell](http://uhaweb.hartford.edu/irussell)

**Office:** Dana 343

**Email:** [irussell@hartford.edu](mailto:irussell@hartford.edu)

**Office Hours:**

Mondays & Wednesdays: 1:10 p.m. – 2:55 p.m.

Thursdays: 2:00 p.m. – 3:00 p.m.

Other times by appointment

## Course Description:

The study of linear lists (stacks, queues); static versus dynamic allocation; garbage collection; sparse matrices; prefix, postfix and infix formulas; recursion and recursive algorithms; tree (binary and other representations of trees, traversal of trees); hashing; searching; sorting (bubblesort, quicksort, heapsort, mergesort); analysis of algorithms in terms of time and space complexity; graphs (representation of graphs, topological sorting, reachability, critical path algorithm). Laboratory Fee.

**Course Prerequisites:** CS 115 with a minimum grade of C

## Learning Outcomes:

- To have an understanding of basic algorithmic analysis and be able to identify the differences among best, average, and worst case behaviors.
- To understand the Big-Oh notation and the concept of time and space tradeoffs in algorithms.
- To be able to apply program design techniques and software engineering principles to building a medium-sized system, with algorithmic efficiency in mind.
- To be able to implement linear search and binary search algorithms and discuss several sorting algorithms and their complexities.
- To understand the underlying concepts of recursion and be able to implement recursive functions and recursive backtracking.
- To understand and be able to apply various implementations of fundamental data structures including linked lists, stacks, queues, hash tables, and graphs and trees and be able to select an appropriate data structure for a given problem.
- To understand and be able to implement fundamental computing algorithms involving hash tables, binary search trees, depth-first and breadth-first traversals, shortest path algorithm, minimum spanning trees, and heaps.

## **Textbook (Optional):**

*Data Structures and Algorithms in Java*, 5th edition, Goodrich and Tamassia, Wiley publisher.  
<A copy of the book is available on reserve at the library>

## **Facilities and Software:**

We will be using the computer science teaching lab located in Dana 318. Several Java development environments are available in D 318 and also in the computer science open lab in Dana 230. Students will need an account on the system. Below is download information on various IDEs. You may use any Java environment that you feel comfortable with.

**BlueJ version 3.0.5** available at:

<http://www.bluej.org/download/download.html>

### System Requirements:

You need to have Java 6 or Java 7 (also known as JDK6/JDK7) installed on your system. Get it here: [JDK download](#). Note that only the version labeled **JDK** will work. Not the one labeled **JRE**, nor the one that comes with NetBeans, nor the EE version.

**BlueJ Tutorial** is at: <http://www.bluej.org/tutorial/tutorial.pdf>

### **Eclipse Java:**

<http://www.eclipse.org/downloads/download.php?file=/technology/epp/downloads/release/galileo/SR1/eclipse-java-galileo-SR1-win32.zip>

**Netbeans IDE:** [download it here](#)

**NetBeans/BlueJ IDE:** is at: <http://edu.netbeans.org/bluej/>

## **Exams:**

Test 1	Wednesday October 5, 2011
Test 2	Wednesday November 16, 2011
Final Exam	Friday December 16, 2011 2:00 p.m. – 4:00 p.m.

## **Grading:**

Projects	55%
Exams	45%

The 2011-2012 **academic calendar** is available at: <http://www.hartford.edu/newsevents/calendar/academic.asp>

## Readings:

- Chapter 1:  
Java  
Programming  
Basics** Classes, types, objects, methods, control flow, arrays, input/output
- Chapter 2:  
Object-Oriented  
Design** Overview of object orientation, objects, classes, encapsulation, interfaces, inheritance, polymorphism, exceptions, interfaces, abstract classes, casting and generics.
- Chapter 3:  
Indices, Nodes,  
Recursion** Using arrays, singly linked lists, doubly linked lists, circular linked lists, recursion.
- Chapter 4:  
Analysis Tools** Comparing growth rates, analysis of algorithms.
- Chapter 5:  
Stacks, Queues,  
Deque** The Stack Abstract Data Type, array-based stack implementation, implementing a stack with a generic linked list, the Queue Abstract data type, array-based queue implementation, implementing a queue with a generic linked list, double-ended queues.
- Chapter 6: Lists,  
Iterator ADTs** Array lists, node lists, iterators, List ADTs and the Collections Framework
- Chapter 7:  
Trees** Implementing trees, tree traversal algorithms, implementing and using binary trees.
- Chapter 8: Heaps,  
Priority Queues** The priority queue ADT, implementing a priority queue, heaps.
- Chapter 9: Hash  
Tables, Maps, skip  
lists** The Map ADT, Hash tables, the dictionary ADT.
- Chapter 10:  
Search Trees** Binary Search Trees, AVL trees, Splay Trees, Red-Black trees.
- Chapter 11:  
Sorting, Sets, and  
Selection** Merge sort, Quick-sort, Bucket-sort, Radix-sort, comparison of sorting algorithms, the Set ADT and Union/Find Structures.

## **Course Policies and Procedures:**

### **Homework**

I strongly urge you to start working on a program as soon as it is assigned. You will find out that it will take longer than you think. Assignments are due no later than 2:55 p.m. on the due date and should be submitted as a zip file via Blackboard's assignment tool. The penalty for late assignments is 10% between Monday and Wednesday and 15% between Wednesday and Monday. Assignments will be given a grade of zero after one week. However, you are encouraged to complete and submit a late assignment even if it is past one week.

### **What to Submit**

Submit your work via Blackboard's assignment tool.

You should submit a zip folder with your java project. Your project must be well documented. Make sure to run the `javadoc` program to obtain a prettily formatted version of your documentation in HTML format.

Instructions on how to use Blackboard are available at: <http://uhaweb.hartford.edu/fcld/handoutsstudent.htm>

The folder should also have a document that includes:

- (1) Your name, instructor's name, date created and last update.
- (2) Documentation, test cases and reasons for selecting them, screen shots, a summary of the results of your testing, and a paragraph reflecting on your work with comments on how you would do the assignment differently if you were to do it again.

You are encouraged to seek feedback on your work on the project from me, and from the computer science tutors (schedule posted on Blackboard).

Once you submit the project, you are required to meet with me to demo your project.

### **Attendance and Participation**

You are expected to attend every class. You should inform me of any absences prior to class. You will have opportunities for in-class extra credits. You will not be able to take advantage of this if you are not in class.

### **Blackboard**

Assignments and a select course material will be posted on blackboard ([blackboard.hartford.edu](http://blackboard.hartford.edu)). You are expected to logon to the system on a regular basis, at least once a day. I will announce in class when an assignment is posted on Blackboard but it your responsibility to keep track of due dates.

You are not to rely completely on blackboard. Some material as well as some announcements given in class will not be announced on blackboard. You are responsible for all material and announcements given in class including those that are not on Blackboard.

### **Plagiarism**

Plagiarism is representing someone else's work as your own. You are not to copy programs or other work from any source including books and the internet without proper citation. You are expected to submit your own work. You may discuss an assignment with students and certainly with me, but you are expected to submit your own work on all assignments. Any assignments that are copied will be given a grade of zero as well as the ones that were copied from. The students involved will be subject to further disciplinary action. The academic honesty policy is described in the student handbook, the Source, which is available online at

<http://www.hartford.edu/thefsource>.

**Students with Special Needs**

I strongly encourage students with special needs, whether an athlete or a student with a documented learning disability, to let me know of their needs as soon as possible so I can accommodate them.

**Make-up Exams**

Make-up exams are only given when there are documented extenuating circumstances and only if they are pre-arranged.

**Office Hours**

I have regular office hours. They are listed in the syllabus and are also posted on my web page. I urge you to make use of them as soon as you need help. Do not wait until it is too late. You do not need to schedule an appointment to see me during my office hours. Feel free to drop by anytime even to just say hello. You may also meet with me at other times by appointment.

**Tutoring lab**

The department of computer science offers free tutoring in Dana 230. No appointment is needed. I urge you to use this service. I will distribute the tutoring schedule once it is available. The tutoring lab is staffed by full time faculty members as well as upper class students.

**Cell Phones**

Cell phones must be turned off during class.