4003-233 | **Computer Science 3** | **Handout 1**
Term 20092 | **November 30, 2009**

Syllabus

General Information

Instructor: Matthew Fluet
E-mail: `mtf@cs.rit.edu`
Office hours: M 11am – 12pm & 1pm – 2pm, W 11am – 12pm & 4pm – 5pm (or by apt.); 70-3555

Lectures:
Section 01 MW 9:00am – 10:50am; 70-1400
Section 03 MW 2:00pm – 4:50pm; 70-1400

Labs:
Section 41 R 8:00am – 9:50am; 70-3640 (ICL 5) (Fluet)
Section 42 R 8:00am – 9:50am; 70-3620 (ICL 6) (Hicks)
Section 43 F 10:00am – 11:50am; 70-3520 (ICL 1) (Fluet)
Section 44 F 10:00am – 11:50am; 70-3540 (ICL 2) (Patel)

Home page:
`www.cs.rit.edu/~mtf/teaching/20092/cs3`
`www.cs.rit.edu/~vcss233`
`mycourses.rit.edu`

Note: This syllabus document summarizes (and, in minor ways, extends) the general Computer Science 3 syllabus (`http://www.cs.rit.edu/~vcss233/Syllabus/syllabus.html`). You are required to be familiar with that document in addition to this one.

Course Description

This course is the third course in the computer science introductory sequence and builds upon the computer science foundations and design principles presented in Computer Science 1 and Computer Science 2. Students will learn how to use linear data structures, such as stacks, queues, and lists and non-linear data structures, such as trees and graphs, and will also be introduced to the design and analysis of algorithms. Students will learn how to analyze the efficiency of basic sorting, searching, and hashing algorithms, and acquire an understanding of how recursion works. Object-oriented programming will be used to design solutions and implement them as Java programs. Programming assignments — labs and projects — are an integral part of the course.

Prerequisites

- 4003-232 (Computer Science 2)

Course Goals

The goal of this course is to build upon and extend the Computer Science foundations presented in Computer Science 1 and Computer Science 2. This course intends to give the student the knowledge and ability to implement efficient realizations of commonly used algorithms and data structures, as well as a deeper understanding of algorithmic design principles.

Text Books

Required:

Title:	Data Structures & Algorithms in Java (Fourth Edition)
Author:	Michael T. Goodrich and Roberto Tamassia
Publisher:	John Wiley & Sons, Inc.
ISBN:	978-0-417-73884-8
Home page:	`http://java.datastructures.net`

Suggested:

Title:	Introduction to Algorithms (Second Edition)
Author:	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein
Publisher:	MIT Press
ISBN:	978-0-262-53196-2

Topics

- Linear Data Structures
- Complexity Analysis
- Search Algorithms
- Recursion
- Sorting Algorithms
- Trees
- Heaps
- Graphs
- Hashing
- Ethics

Grading Policy

Grades will be assigned based on the following grading scheme:

Midterm exams (x2):	30.0%
Final exam:	25.0%
Labs (x10):	20.0%
Projects (x2):	20.0%
Attendance & participation:	5.0%

After a graded exam, lab, or project has been returned, you have **one week** to bring any questions about grading to the instructor's attention. No grade adjustments will be made after this time.

Midterm exams

There will be two midterm exams; see below for dates. Midterm exams will be administered during the 2nd hour of a lecture period.

Midterm exams must be taken at their scheduled time. Make-up midterm exams will not be administered, unless exceptional circumstances have been discussed with the instructor **in advance** of the exam date and/or other arrangements have been made.

Final exam

There will be a final exam; see below for the date. The final will be comprehensive and will cover material from the entire course, including lectures, labs and assigned readings.

The final exam must be taken at its scheduled time. Any exam conflicts must be reported to the instructor by the end of Week 6.

Laboratories

Each student should be registered for a laboratory section (in addition to a lecture section). Laboratory sections meet once a week, beginning in Week 1. Each laboratory writeup is posted on the course web page; students must **read the lab writeup** and **complete the pre-lab activities** before the laboratory section.

Completed lab assignments are due 5-days after the corresponding lab section. (For example, students with lab section on Thursday must submit their completed lab assignment before the end of the following Tuesday.) **Late submissions will not be accepted.**

As described in the general Computer Science 3 syllabus document, the laboratory grade is computed with a 5% curve.

Projects

There are two (substantial) programming projects. There will be approximately four weeks to complete each project and each project will require an initial and a final submission; see below for dates. **Late submissions will be penalized.** Both the initial and the final submission will contribute towards the project grade (i.e., the initial submission is required for full credit on the project).

Note: Each project will require significantly more time than a laboratory assignment.

Attendance & participation

Students are required to attend and expected to participate in both lecture and laboratories. (Participation means asking and answering questions, not simply attending.) The use of cell phones and audio players is prohibited during class.

As noted above, pre-lab readings and activities must be completed before the laboratory section. Similarly, assigned readings are to be completed before the lecture section. You are responsible for the material in the assigned readings, whether covered during lecture or not.

Important Dates

December 20 (Sun.):	Project 1 initial submission
January 6 (Wed.):	Mid-term exam I (in class, 50 min)
January 17 (Sun.):	Project 1 final submission
January 31 (Sun.):	Project 2 initial submission
February 3 (Wed.):	Mid-term exam II (in class, 50 min)
February 21 (Sun.):	Project 2 final submission
February 25 (Thu.):	Final exam (2:45pm – 4:45pm; 01-2000; Sections 01 & 03)

Academic Integrity

As with all courses, the RIT Honor Code and the RIT Academic Honesty Policy apply. See the Department of Computer Science's statement on academic integrity for more details.

In this course, all submitted work must be your own work (i.e., written up or programmed by you alone) and must include acknowledgments of any collaborators or sources (other than course text books or handouts) used to produce your submission.

Disclaimer

I reserve the right to make any changes to the syllabus as I deem necessary throughout the course. Minor changes, such as programming project due dates, will be announced orally during class and posted on the course mailing list and home page. Major changes, such as grading percentages, will additionally be provided in writing.

C Programming Tutorial Java Tutorial Inheritance in Java Top Java Projects you need to know in 2020 Java Interview Questions What is the use of Destructor in Java? Polymorphism in Java Multithreading in Java All you Need to Know About Implements In Java Spring Interview Questions PHP Tutorial PHP Interview Questions Python Tutorial Python Interview Questions VIEW ALL. Project Management and Methodologies.Â C Programming and Data Structures (24 Blogs). Comprehensive Java Course (2 Blogs). Java/J2EE and SOA (322 Blogs). Spring Framework (8 Blogs). SEE MORE. Top Data Structures & Algorithms in Java That You Need to Know. Last updated on Nov 28,2019 15.3K Views. Archana Choudary.