

CST 250 – Computer Science II – Spring 2010

Michael Ruth, Ph.D.

Course Meetings: T/Th, 3:30pm to 5:35am @ Gage Bldg, Rm. 208

Office(s) & Hours: (Also by appointment – email for time/location)

SCH 600WW: M/W 11am – 12pm, W 5pm – 6:30pm | **Gage, Rm 506A:** T/Th 1:30pm to 3pm

Email: mruth@roosevelt.edu

Phone: (312) 281-3303

Course Description:

Continuation of CST 150; development of problem solving using a high level language including abstract data types, multidimensional arrays; strings; records and structures; function design, construction, and parameter passing methods; recursion; introduction to objects and classes; pointers: lists, stacks, queues, and trees; and file I/O. Multiple sorting and searching algorithms including concepts of program complexity. Extensive programming required. A computer use course.

Course Prerequisites:

CST 150 with a min grade of C-

Course Objectives:

- Illustrate the use of object-oriented concepts such as Encapsulation, Composition, Inheritance, and Polymorphism to solve real world problems in Java.
- Demonstrate mastery in intermediate programming skills such as catching and handling errors, software testing, generic programming, and introductory algorithm analysis.
- Examine fundamental data structures used in Computer Science (including linear and non-linear) and the use of the Java Collections Framework in Java development.
- Implement a subset of fundamental data structures, linear and non-linear, and classic algorithms, such as sorting and searching, to solve real-world problems using Java.
- Determine which data structures and algorithms are appropriate for various application requirements and explain the trade-offs which exist depending on the choices made.

Textbook:

Data Structures in Java: From Abstract Data Types to the Java Collections Framework, by S. Gray – ISBN: 978-0321392794

Grading (+/- grading is used)

- 2 Exams (15% each)
- Final Exam (20%)
- Class Participation/Assignments (50%)

Academic Honesty:

Any instance of academic dishonesty will result in a **zero** grade on that assignment/exam. Any second instance will result in an 'F' in the course. Each assignment/exam must represent **your own work**. You may discuss assignments with other students, but you cannot share any assignment artifacts. The RU grievance procedure can be found online at: [http:// roosevelt.edu/current/judicial/](http://roosevelt.edu/current/judicial/)

Disabilities:

If you have a disability for which you are or may be requesting an accommodation, you are encouraged to contact the Office of Disabled Student Services, 310 Herman Crown Center, 312-341-3810, or e-mail nlitke@roosevelt.edu as early as possible in the term.

Course Policies:

- You only have **three free** absences. After that, you will be penalized a letter grade for each additional absence. ***Make them count!***
- ***You are responsible*** for all material covered and announcements regardless of delivery.
- ***There will be no make-up examinations:*** If an exam is missed ***due to an emergency***, an exam of ***my choice*** will count for both!
- Late homework will be accepted, with or without penalties, at my discretion.
- Arrive in class ***on time*** and ***silence all noise-producing equipment!***

Tentative Course Schedule:

Date	Topic	Reading
1/26	Introduction to CST 250/Diagnostics	0
1/28	OOP (Inheritance/Composition)	1.1-1.3
2/2	Polymorphism & Generic Programming	1.4-1.6
2/4	Error Handling	2.1
2/9	Testing Software	2.2
2/11	Algorithm Analysis/Algorithm Measurement	2.3
2/16	Fundamental Data Structures: Arrays	3.1-3.2
2/18	Fundamental Data Structures: Linked DS	3.3-3.4
2/23	Linked Data Structure Variations	3.5-3.7
2/25	Midterm	
3/2	Collection Classes/Iterators	4.1-4.4
3/4	More Collection Classes, Implementation, and Analysis	4.6-4.9
3/9	List ADT	5.1-5.4
3/11	List ADT (Implementation, testing, and analysis)	5.5-5.7
3/23	Stack & Queue ADTs	6.1-6.4,7.1-7.4
3/25	Implementation, Testing, & Analysis of Stack & Queues	6.5-6.8, 7.5-7.8
3/30	Recursion	8
4/1	Introduction to Sorting & Searching	9.1
4/5	Last Day To Drop With a "W"	
4/6	Sorting & Searching II (Selection, Insertion, Gnome Sorts)	9.2
4/8	Midterm	
4/13	Sorting & Searching III (Recursion - QuickSort/Mergesort)	9.3-9.5
4/15	Sorting & Searching III (Objects)	10.1-10.3
4/20	Non-Linear Data Structures: Trees	10.4
4/22	Tree ADT (I, T, and A)	10.5
4/27	Heaps and Binary Trees	11.1
4/29	Binary Search Trees	11.2
5/4	Binary Search Tree Implementation	11.2
5/6	More Advanced Data Structures	11.3-11.5
5/13	Final Exam	