Lecture 1: Course Overview

Instructor: Olga Veksler
Course Content

- Study useful data structures and algorithms that are essential in core areas of computer science
  - Compilers
  - Operating Systems
  - Database Systems
  - Search Engines
  - etc.

- This is one of the most important courses you will take
  - prerequisite for almost all other upper year courses
  - Google/Microsoft/etc interview questions
Data Structures you already Know

- Data Structure is representation and operations associated with a data type.

Set/bag

Array

Tree
Basic Algorithms you already Know

- Binary search
- Bubble sort
- Quick sort
Knowledge Assumed

- We assume you know basic data structures
  - arrays, bags, lists, stacks, queues, linked lists
- Functions, logarithms, exponents, sets, series, derivatives, limits
- Abstract Data types
- Basic object oriented design concepts
  - abstraction, encapsulation, modularity
Skills Assumed

- **We assume you know how to program in Java**
  - can design, implement, test, debug, read, understand and document relatively simple Java programs
  - including simple recursive code
- **This is not** a course on Java. If you don’t know Java, start learning it now, before the programming assignments are due!
  - books, tutorials on the web, etc.
  - chapters 1,2 of our text book (Goodrich&Tamassia) gives a good introduction to Java
What will you learn?

- Analysis of moderately complex algorithms
  - How to predict algorithm’s performance
    - time and space complexity
  - Algorithm correctness

- Widely useful data structures that let us efficiently store, access, manage data

- How to choose appropriate data structures and algorithms for a given application
Topics Covered

- Analysis of algorithms
  - Time and space complexity
  - Correctness
- Data Structures
  - Dictionaries, hash tables
  - Priority queues and heaps
  - Trees, binary search trees, multi-way search trees
  - Graphs
- Algorithms
  - Binary Search, sorting, algorithms on trees and graphs
- Java Topics: Interfaces
Contact Info

- **Instructor Information**
  - Olga Veksler
  - Lectures: Mon 3:30-5:30, Th 4:30-5:30
  - Office hours im MC361
    - Tues 3:30-4:30
    - Thurs 3:30-4:30

- **Lecture Notes**
  - Available on the course web page
Contact Info

- Course Web Page
  - http://www.csd.uwo.ca/courses/CS2210b

- You are responsible for checking the course web page frequently

- I will post any important announcements on the web page (about homework, exams, etc.)
Course Discussion Board

- Link on the course web page
- Use it to ask any questions about the course
  - You can post anonymously
  - You should also feel free to reply to other student’s questions
  - Can’t post code, but one liners are ok
- If you have a question about homework
  - Post it on the discussion board
  - Come talk to me or the TA
  - Please do not email me about homework, unless it is to set up a meeting outside the office hours
Course Prerequisites

- Computer Science 1027a/b or 1037a/b with grade of at least 60%
- One full-course equivalent from
  - Calculus 1050a/b, 1051a/b, 1081a/b, Linear Algebra 1040a/b, Applied Mathematics 1026, Mathematics 030, the former Mathematics 1027, 1023a/b, 1024a/b, the former Mathematics 1027
- Knowledge of Java
Textbook


- **Supplementary**:
  - *Data Structures and Algorithms with Object Oriented Design Patterns in Java*, Bruno R. Preiss, Wiley, 2000
  - Other books listed on the course web site
Student Evaluation

- Assignments 36%
  - 2 concept assignments, not graded, -2% if not handed in
  - 3 programming assignments, each worth 12%

- Midterm 29%, 2 hours
  - February, during class
  - Final Exam 35%, 3 hours

- If an assignment has to be cancelled, the remaining assignment weights will be readjusted, if the midterm has to be cancelled, final exam will be worth 64%
Student Evaluation

To be eligible to pass this course need
- 45% weighted average on the midterm and final exams \textbf{and}
- 45% weighted average on assignments

To be eligible for an overall grade of 60% or higher, need
- 55% weighted average on the midterm and final exams \textbf{and}
- 55% weighted average on assignments
Tentative Assignments Schedule

- Assignment 1 (concept) due January 18
- Assignment 2 (programming) due February 7
- Assignment 3 (concept) due February 15
- Assignment 4 (programming) due March 16
- Assignment 5 (programming) due April 5
Submission Procedure

- Paper copy of assignments are due in class on the due date, or electronically on webct.
- Programming assignments are to be submitted only electronically on webct.
Exam Policies

- No midterm makeup, except for religious reasons
  - notify the course instructor and fill documentation with Dean’s office at least 2 weeks prior to the Midterm Exam

- If you miss Midterm for any other valid reason, present documentation to the Dean’s office
  - your Final exam score will be re-weighted to include the weight of the Midterm
Class Attendance

- Not required, but strongly encouraged
  - Lecture notes are only a summary
  - Students who attend lectures tend to do better
  - Attending lecture is most efficient way to learn the required material
Assignments

- Posted on the web site
- Concept Assignments
  - a set of exercises
  - must be typed or neatly written
- Programming assignments
  - Due by 11:55 pm on the due date
  - Java programs
  - electronic submission via OWL
Late Policy

- Concept Assignments
  - No late concept assignments accepted

- Programming assignments
  - 1 day late: 5%
  - 2 days late: 10%
  - 3 days late: 15%
  - 4 days late: 20%
  - 5 days late: not accepted
  - Saturday and Sunday count as one day
Late Policy

- Assignment extensions may be granted only by the course instructor

- If you have serious medical or compassionate grounds for an extension, take the supporting documentation to the Dean of your faculty
  - they will contact me
Assignment Marks Appeals

- Address appeals to the TA first
- If you and the TA cannot agree, TA will discuss the situation with the instructor
- Appeals must occur within 1 week from the first day that the marked assignments were made available to students
Ethical Conduct

- Assignments are to be done individually
- Assignments judged to be the result of academic dishonesty will,
  - first offence = -assignment weight
  - second offence = failing mark
- You are responsible for reading and respecting the Computer science Department’s policy on Scholastic Offences
- We use sophisticated software that will examine your code against everyone else in class