UNIVERSITY OF ALBERTA

Augustana Computing Science 112
Data Structures & Algorithms

Lecture A1; Labs H1, H3, H4
Winter 2025

Instructor: Rosanna Heise

Office: C 1-108 (Classroom Building, NW corner)
Office Hours: Tuesday/Thursday 12:00 noon – 1:00 p.m.
            Wednesday 12:30 p.m. – 1:30 p.m.
            Or by appointment (send email)
Phone: 780-679-1674
Email: rosanna@ualberta.ca

Welcome to Java Programming!

Calendar Course Description:

An introduction to object-oriented design and programming in Java; algorithm analysis; data structures and container classes including arrays, lists, arraylists, stacks, queues, user-defined structures and an introduction to trees, along with associated algorithms such as insertion, removal, sorting and retrieval; introduction to recursion.

Prerequisite: One of AUCSC 111, AUCSC 113 (2023), or AUSCI 135 (2023) *Introduction to Computational Thinking*.

Corequisites: AUMAT 116 *Calculus* and AUMAT 120 *Linear Algebra*.

Course Objectives:

1. To create substantial computer programs in a new programming language, in order to understand general principles of computing (versus principles of one programming language).
2. To use simple concrete programming concepts to create a variety of abstract structures (arrays, lists, stacks, queues, trees) and ways to manipulate them.
3. To effectively use pre-made abstract structures in programs.
4. To determine the costs and trade-offs (primarily time and space) of program code, and to program in a manner that reflects this knowledge.

Core Skill Development:

We focus on the Thinker and Communicator aspects.

Thinker

- Take natural language and develop complete computer programs.
- Precisely determine how to put together computational concepts.

Communicator

- Convey (clearly, orally) computational ideas within the class, and in discussions with the instructor.
- Hone written skills for expressing computational ideas, through the writing of program code and also English documentation to enhance that code.

Class Times:

Tuesday/Thursday 8:30 – 10:00 a.m. in C 1-115 (NE corner of the top floor of the classroom building)

Lab Times: always bring your laptop and charger to one of

Lab H1: Tuesday 2:15 – 5:15 p.m. in C 1-115.
Lab H3: Thursday 2:15 – 5:15 p.m. in C 1-115.
**Lab H4:** Wednesday 2:15 – 5:15 p.m. in Forum 1-130.

Class Website / Past Evaluative Material:

Assignments, marks, past evaluative material and some notes will be available on eClass. In particular, there are copies of all slides used during class and it is recommended that students use these to write on during lectures or labs.

Textbook: (available online)

Heise, Rosanna. *Practices in Data Structures and Algorithms with Java*. 2023. [https://sites.ualberta.ca/~rosanna/books/PDSA]
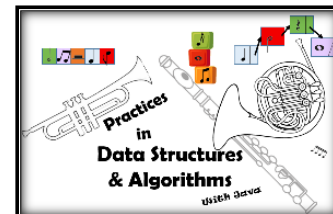
Mark Distribution:

Assignments ……………………………………..………………24%

All assignment specifications can be found on eClass. Students must attend labs, complete the check-in and check-out, and be working on the assignments during the designated lab time to receive credit for an assignment. Furthermore, students must be able to answer questions regarding their assignment submission, to receive credit. Credit may be prorated, if students submit solutions they cannot explain properly – see also the section on Academic Integrity below. There will be eight assignments. No generative AI or homework help sites may be used.

Mid Lab Exam (Mar. 4/5/6) ………………………………………….. 6%

Will be completed on students' own computers during the lab, using the IDE and Java.

Final Lab Exam (Apr. 8/9/10) ..………………………………………10%

Will be completed on students' own computers during the final lab, using the IDE and Java.

Quizzes ……………………………………………………… 5%

Will occur at random times during class. Students need to have their computers, with ExamLock installed and working.

Paper Coding ……………………………………………………… 5%

Will occur at random times during class. Students need to have a pencil and eraser.

Midterm Exam (Mar. 13).......….....………….…….…..…………… 15%

This will happen during the scheduled class time, with pencil/pen and paper.

Final Exam (Apr. 28)…………..….…………………………...………35%

The date is scheduled by the Registrar, during the final exam period, written in the gym.

Bonus Marks Available …………………………….. up to a maximum of 5%

If you find an error in the textbook, you may email the instructor to claim: 0.5 per syntax/formatting error; 1 per conceptual error. The instructor must agree that there is an error as you describe.

Determination of Final Grades:

Copyright © 2002 Newspaper Enterprise Association, Inc.

Using the above evaluation scheme, a percentage, final mark will be determined. This will be converted to the U of A's four-point system, using the distribution guidelines, *approximately* as follows:

Grade		Descriptor	Percentage
A	+/-	Excellent	83—100
B	+/-	Good	70—82
C	+/-	Satisfactory	60—69
D	+	Poor	50—59
F		Failure	0—49

Academic Integrity:

This is serious – pay attention! The University of Alberta is committed to the highest standards of academic integrity and honesty. Students are expected to be familiar with these standards regarding academic honesty and to uphold the policies of the University in this respect. Students are particularly urged to familiarize themselves with the provisions of the Code of Student Behavior and avoid any behavior which could potentially result in suspicions of cheating, plagiarism, misrepresentation of facts and/or participation in an offence. Academic dishonesty is a serious offence and can result in suspension or expulsion from the University. (GFC 29 SEP 2003) Another guide that should be consulted is the Undergraduate Students' Guide to Academic Integrity.

Instead of taking a chance, please ask for help!

All of the following, and potentially more, constitute behavior that is not honest and should NOT occur in this class:

Careful… Do NOT do this!

- o Submitting someone else's work as your own, in whole or in part
- o Creating a work with the knowledge that someone else will use it fraudulently (whether you are paid or not paid)
- o Altering someone else's work (e.g. by changing variable names or moving the order of functions) to conceal the fact that it is not your own
- o Using portions of another's work (including algorithms, code, or answers from a textbook, magazine, internet, or other sources) without acknowledgement
- o Typing in code or answers that are identical, or almost identical, to another student
- o Using online help sites, including submitting assignments (all assignments are the property of the instructor(s) of the course), or quiz or exam questions
- o Using an AI (Artificial Intelligence/computer program) to generate code, text, or answers for a graded assignment, quiz or exam, or providing assignments, or quiz or exam questions to an AI

chatGPT

Students are encouraged to help one another in completing assignments at two levels:
- o Conceptually – explain concepts to one another, explain programming techniques…do not discuss exact lines of code
- o Finding bugs – help each other solve compile time errors or obscure run-time errors

Recording of Class Sessions or Any Part of a Class Session:

Audio or video recording of lectures, labs, seminars or any other teaching environment by students is allowed only with the **prior written consent** of the instructor or as a part of an approved accommodation plan. In addition, if a student wants to take a photograph of the screen, permission must be requested of the instructor. Recorded material is to be used solely for personal study, and is not to be used or distributed for any other purpose without prior written consent from the instructor

Student Accessibility Resources:

Students registered with Accessibility Resources who will be using accommodations in the classroom, or who will be writing exams through SAS, are required to provide an "Accommodation Letter" to the instructor as soon as possible. Make an appointment with the instructor to discuss these accommodations and any other related concerns. If you require accessibility resources, please contact the advisors at 780-679-1649, or augar@ualberta.ca.
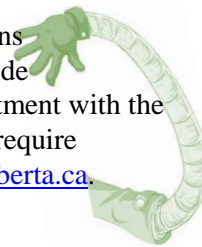
Investment of Time/Tuition:

It is recommended that full-time students take on no more than 10 hours of part-time work per week. Although it may be tempting to spend more time working and less time studying, it is important to remember that the money invested in tuition will be lost if you fail the course.

Absence from Class:

Please notify the instructor if you will be absent from a lecture or lab. Note that absences have an effect on your grade (see Section on Mark Distribution, above).

Cell Phone and Mobile Computer Use:

Our class and lab times are reserved for learning…as such, do not text, send emails, answer phone calls, watch videos or engage in other distracting behavior during these times. Use your phone only in ways that are helpful to learning course material. Note that using mobile devices may have an effect on your grade (see Section on Mark Distribution, above).

Computer:

Students are expected to have a computer, and bring it, charged, to each class. The department has a limited number of laptops available to loan to students in cases of emergencies. Please contact the instructor if you need to borrow one.

Programming Language and Integrated Development Environment:

We will use the Java programming language, and IntelliJ IDEA Community Edition. Please come to the first lab with both installed.

Java 23 (you may need a free Oracle account)
https://www.oracle.com/ca-en/java/technologies/downloads/
"JDK Development Kit", pick the tab with the correct operating system (the default is Linux, and you will likely want either macOS or Windows).
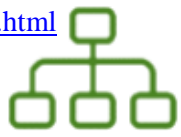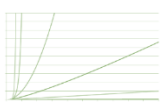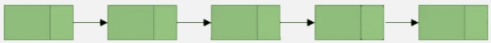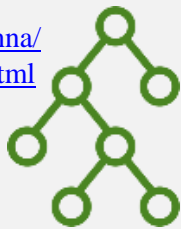
IntelliJ IDEA Community Edition
https://www.jetbrains.com/idea/download/
It is important to select the "Community" edition (not the "IntelliJ IDEA Ultimate"). The community edition is further down on the page, so scroll down. You may not use the professional version in this class.
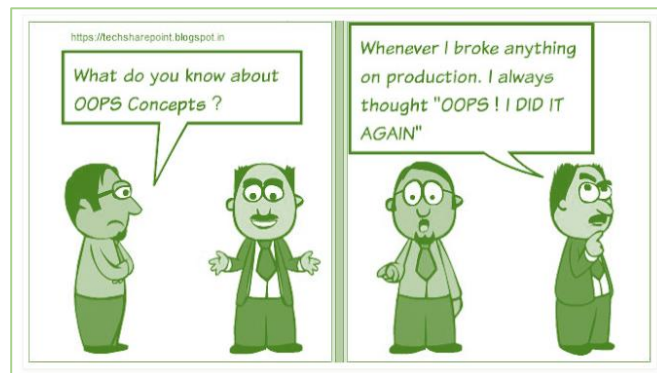
Policy:

Policy about course outlines can be found in Course Requirements, Evaluation Procedures and Grading of the University Calendar.

Brief Outline:

Week #	Topic	Read
1 Jan. 28/29/30	**Introduction to the course and to Java.** How to use Intelli-J. First programs in Java. Algorithms. Variables, text and arrays (1D). Simple sorting with visualization. Primitive and reference data types. A conditional (if). Two ways of repetition (for-each, while). Methods. **Lab 1 – Introduction to Java (Three Small Programs)**	Heise – Chapters 0 & 1 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter0/intro.html https://sites.ualberta.ca/~rosanna/books/PDSA/chapter1/intro.html **Be sure to install Java and IntelliJ.**
2 Feb. 4/5/6	**More structure.** Arrays (multidimensional). Repetition on structures, including while loop. Multiple files. User input. **Lab 2 – Arrays and Classes (Sinking Ship)**	Heise – Chapter 2 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter2/intro.html
3 Feb. 11/12/13	**Hierarchies and OOPs.** Inheritance. Setters and getters. Constructors. String representation. Abstract methods. **Lab 3 – Object-Oriented Programming with Hierarchies (Banking)**	Heise – Chapter 3 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter3/intro.html
4 Feb. 18/19/20	**Reading Week – No Classes or Labs**	
5 Feb. 25/26/27	**Complexity analysis.** The seven functions (constant, logarithm, linear, $n$log(n), quadratic, cubic, exponential) **Lab 4 – Complexity Analysis (Worksheet)**	$O(n)$ Heise – Chapter 4 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter4/intro.html
6 Mar. 4/5/6	**Linked Lists.** Node. Data fields. Operations. Complexity comparison with arrays. Doubly linked and circular lists. **Mid Lab Exams**	Heise – Chapter 5 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter5/intro.html
7 Mar. 11/12/13	**Catch up** **Midterm Exam on Mar. 13** **Lab 5 – Linked Lists (Hot Potato)**	
8 Mar. 18/19/20	**Abstract Data Types (ADTs).** Stacks. Queues. Deques. **Lab 6 – Stacks (Battlecards)**	Heise – Chapter 6 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter6/intro.html

9 Mar. 25/26/27	**Recursion.** Base case. Wrapper. **Lab 7 – Recursion and Interfaces (Triangle and Kind of Sort)**	Heise – Chapter 7 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter7/intro.html
10 Apr. 1/2/3	**Quicksort.** Comparison to other sorts. Partitioning both in general and in place. **Introduction to binary trees.** Node. Data fields. Traversals (depth-first, breadth-first, pre-order, post-order, in-order, Euler). Search trees. **Lab 8 – Binary Search Trees (Documentation to Add and numGreaterThan method)**	Heise – Chapter 8 https://sites.ualberta.ca/~rosanna/books/PDSA/chapter8/intro.html
11 Apr. 8/9/10	**More binary trees. Arraylists.** How to use. Space and time complexity issues. **Final Lab Exams**	Heise – Chapter 8 0 1 2 3 4
12 Apr. 15	**Catch Up and Review.** **Apr 15 is our last day of class**	
13 Apr. 28	**Final Exam** – 9:00 am in gym, date set by registrar	