

Course Summaries: UW Computer Science

CSE143

Course Description

This course is a continuation of CSE 142. While CSE 142 focused on control issues (loops, conditionals, methods, parameter passing, etc.), CSE 143 focuses on data issues. Topics include: ADTs (abstract data types), stacks, queues, linked lists, binary trees, recursion, interfaces, inheritance, and encapsulation. The course also introduces the notion of complexity and performance tradeoffs in examining classic algorithms such as sorting and searching and classic data structures such as lists, sets, and maps.

Course Objectives

- Implement and utilize various data structures such as stacks, queues, linked lists, and binary trees.
- Understand and apply recursion in problem-solving.
- Analyze the complexity and performance tradeoffs of algorithms.
- Utilize Java Collections Framework components effectively.

Topics

- Abstract Data Types (ADTs)
- Stacks and Queues
- Linked Lists
- Binary Trees
- Recursion
- Interfaces and Inheritance
- Encapsulation
- Algorithm Complexity and Performance Tradeoffs
- Sorting and Searching Algorithms
- Java Collections Framework

Course Summaries: UW Computer Science

CSE121

Course Description

Computing continues to play an ever-increasing role in today's society. Having an understanding of computing is an essential skill for those in the 21st century. This course is a first course in computer programming focused on the basics of procedural programming in Java. It assumes students have not taken a previous programming course and do not have substantial other programming experience.

Course Objectives

- Understand the basics of procedural programming in Java.
- Gain familiarity with control structures such as loops and conditionals.
- Develop an understanding of data types, expressions, and methods.
- Implement programs that use arrays and object-oriented concepts.
- Learn debugging and program design through practice-based learning.

Topics

- Course Policies; Hello World
- Printing; Strings; Variables
- Datatypes; Expressions
- Revisiting Strings & Variables
- `for` Loops
- Nested `for` Loops; `Random`; `Math`
- Methods; Parameters; Scope
- Boolean Expressions; if/else
- While Loops
- Arrays
- Classes and Objects
- File Input and Output
- Final Review

Course Summaries: UW Computer Science

CSE122

Course Description

This is a second-level programming course focused on the use of data structures and object-oriented programming. It assumes students already have experience with control structures (loops and conditionals), variables, arrays, and functions. Java is the language of instruction, but prior experience in Java is not required.

Course Objectives

- Solve complex problems by decomposing them into smaller tasks.
- Use data structures such as lists, dictionaries, and sets efficiently.
- Design object-oriented programs emphasizing abstraction between interface and implementation.

Topics

- Functional Decomposition & Design
- Java Review
- File Input/Output (Part 1 & 2)
- ArrayLists and Their Applications
- Object-Oriented Programming
- Nested Collections
- Sets and Maps
- Reference Semantics
- Final Review and Practice Exams

Course Summaries: UW Computer Science

CSE123

Course Description

This is a third course in computer programming focused on the design and implementation of data structures. It assumes students already have significant experience with programming and understand fundamental computing concepts such as control structures (loops and conditionals), variables and data, and arrays, as well as basic data structures such as lists, sets, stacks, queues, and maps/dictionaries, and the fundamentals of object-oriented programming.

Course Objectives

- Implement compound data structures such as lists and trees.
- Use recursion to solve computational problems efficiently.
- Utilize inheritance to represent relationships among similar classes.

Topics

- Inheritance & Polymorphism
- Abstract Classes
- Implementing Data Structures
- ArrayIntList & Linked Nodes
- Linked Nodes with Loops
- Stacks and Queues
- Binary Trees and Tree Traversals
- Recursive Tree Algorithms
- Binary Search Trees
- Sets and Maps
- Priority Queues and Heaps
- Final Project and Review

Course Summaries: UW Computer Science