

**Master Course Syllabus**  
School of Engineering and Computer Science  
Washington State University Vancouver  
**CS 122**  
**Data Structures**  
4 Semester Hours  
(3 lecture hours, 3 laboratory hours)

**Catalog Description**

Advanced programming techniques: data structures, recursion, sorting and searching, and basics of algorithm analysis.

**Prerequisite Courses**

- CS 121 with a C or better or
- CS 251 with a C or better

**Prerequisite Topics**

One of the following:

- Imperative programming language fundamental concepts
- Experience designing, coding and debugging small programs written in an imperative programming language

**Measured Course Outcomes**

None

**Covered Course Outcomes**

Students taking this course will:

1. Select appropriate fundamental data structures for solving problems. (WSU Goal 1.)
2. Implement fundamental data structures and algorithms. (WSU Goal 7.)
3. Analyze the space and time complexity of several fundamental algorithms. (WSU Goal 7.)

**Required Textbooks**

- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, *Introduction to Algorithms*, MIT Press.
- M. Goodrich and R. Tamassia, *Data Structures and Algorithms in Java*, Wiley, John & Sons, Inc.

**Reference Material**

None

**Major Topics Covered in the Course**

1. Data structures including strings, linked lists, stacks, queues, binary search trees, and hash tables

2. Recursion
3. Search algorithms (including linear and binary)
4. Sorting algorithms (including merge sort, quick sort and heap sort)
5. Simple complexity analysis of algorithms

### **Projects**

All programming projects and assignments are to be derived and developed by students individually. In this course, students do not perform assignments as members of teams.

Programming Project Area	Weeks
(Small Assignments, See Below)	

### **Design, Implementation and Analysis**

This course requires the student to craft 6-10 correctly functioning computer programs. The requirements for each program will necessitate that the student comprehends programming language concepts and use them to design and implement programs in C or Java. These programs range from 30-50 lines of code at the beginning of the semester, to substantially larger programs at the end of the semester.

The instructor performs analysis of representative problems in class. All student programming assignments require the student to analyze problem requirements. The instructor analyzes problem solutions (both his own and student's) in class.

### **CS2013**

This course provides coverage of CS2013 knowledge areas. Values listed are minimum course hours dedicated to the topic, percentages indicate the fraction of CS2013 knowledge area topics covered (acceptable values are: <25%, 25-75%, >75%, or 100%).

Area	Tier 1	Tier 2	Elective
AL/Fundamental Data Structures and Algorithms	8 (25-75%)	2 (25-75%)	
PL/Object-Oriented Programming	3 (100%)	3 (100%)	
PL/Basic Type Systems	1 (100%)	2 (100%)	
SDF/Fundamental Programming Concepts	6 (100%)		
SDF/Fundamental Data Structures	12 (100%)		
SDF/Development Methods	4 (100%)		

---

Course Coordinator:	Scott Wallace
Last Updated:	May 11, 2018
Syllabus Version Number:	2.0