
Applied Algorithms Section 01

CS 48

Spring 2025 In Person 1 Unit(s) 01/23/2025 to 05/12/2025 Modified 02/10/2025

Contact Information

Instructor: Dr. David Scot Taylor

All course-related correspondence should go to Eliot Hall (christopher.e.hall@sjsu.edu), who is the student instructor.

Office Hours

By appointment only

Online (via Zoom, invite link will be sent in an email)

Send an email to Eliot Hall (christopher.e.hall@sjsu.edu) to schedule an appointment

Course Information

Lecture

Monday, 9:00 AM to 9:50 AM, MH 422

Course Description and Requisites

Creating and implementing algorithms to solve problems. Techniques covered include using built-in collection classes, bitwise operators, modulo operator, and input/output classes. Emphasis on using data structures learned in CS 46B. Students write a Java program every week.

Prerequisite(s): CS 46B and one Java course (with grades of C- or better), or instructor consent.

Letter Graded

Classroom Protocols

Programming assignments (55%)

We will be doing individual programming assignments. THERE ARE NO LATE ASSIGNMENTS. We will review the solution the class after the assignment is due. You will submit your assignment a system called WebCAT. Since unexpected events may arise, two of the weekly assignments can be dropped without penalty. You will be able to submit your code to WebCAT multiple times.

Individual programming assignments are not group projects. If students get help on assignments, even to resolve a minor problem, it must be documented in the code with the name of the person rendering the help and a brief description of the help provided. Extensive help on a project will disqualify the submission.

Failure to document help, or any other forms of cheating will result in a failing grade on the assignment at a minimum and may result in failure of the course. All incidents will be reported to the Office of Student Conduct & Ethical Development. Even in open source, you cannot copy code from one open source project to another without attribution. Sharing solutions with other students, even if it is indirectly through public source repositories, falls under "aiding and abetting".

The University Policy S16-9, Course Syllabi (<http://www.sjsu.edu/senate/docs/S16-9.pdf>) requires the following language to be included in the syllabus:

"Success in this course is based on the expectation that students will spend, for each unit of credit, a minimum of 45 hours over the length of the course (normally three hours per unit per week) for instruction, preparation/studying, or course related activities, including but not limited to internships, labs, and clinical practica. Other course structures will have equivalent workload expectations as described in the syllabus"

IDE Tip Presentation (10%)

IDEs make us much more productive as programmers. In this class, you can use the IDE of your choice. To help others learn to be more effective with IDEs, you will make a 3-5 minute presentation to show an awesome feature of your favorite IDE. Students can choose their topic. Past topics include triggered and conditional breakpoints, advanced source control, automated refactoring tools, step filters, logpoints/tracepoints, debug consoles, and data breakpoints.

Program explanation (5%)

Shortly after the midterm, each student will be expected to explain a solution to a programming problem of their choice to the class. Your explanations should explain how your solution works without simply showing your solution's code. A good explanation will leave the audience with a clear picture of how to implement your solution in code.

Final Examination and in class programming problem (30%)

There will be three programming problems that will be done in class. They will be timed and will be administered in the style of an exam. You will be limited to only use language references (the Oracle Java docs) for these. The problems are chosen to be similar to weekly assignments.

D	$\geq 63\%$ and $< 67\%$
D-	$\geq 60\%$ and $< 63\%$
F	$\geq 0\%$ and $< 60\%$

There is no rounding, but boundary cases count as the higher of the two grades. The instructor reserves the right to make cutoffs more lenient, but not more strict.

University Policies

Per [University Policy S16-9 \(PDF\)](http://www.sjsu.edu/senate/docs/S16-9.pdf), relevant university policy concerning all courses, such as student responsibilities, academic integrity, accommodations, dropping and adding, consent for recording of class, etc. and available student services (e.g. learning assistance, counseling, and other resources) are listed on the [Syllabus Information](https://www.sjsu.edu/curriculum/courses/syllabus-info.php) web page. Make sure to visit this page to review and be aware of these university policies and resources.

Course Schedule

When	Topic	Notes
1/27/25	IO	Scanner, BufferedReader, FileIO
2/3/25	Strings	StringBuilder, String methods, String memory allocation
2/10/25	Binary Search Trees	Insertion, Deletion, Find Postorder, Preorder, Inorder traversals
2/17/25	Math and Bit Operations	xor, or, and, not, shift left, shift right
2/24/25	Arrays	Arrays vs ArrayLists ArrayLists vs LinkedLists Array access Subarrays

3/3/25	Searching	Binary Search implementation Unbounded Binary Search Binary search for closest element
3/10/25	Sorting	Builtin sorts Applications
3/17/25	In Class Problem (attendance required)	1 problem, all of class
3/24/25	Linked Lists	Memory layout Insertion/deletion time Random access time Algorithms on linked lists
3/31/25	Spring Recess	(No assignments)
4/7/25 + 4/14/25	Hashtables	Hash Functions Chaining Hashtable or Array? Types of Maps
4/21/25	Big O notation	
4/28/25	Useful Java Data Types	BigInteger, BitSet, BigDecimal applications
5/5/25	Lambdas, Functions, and streams	Function, Lambda, Consumer, BiConsumer, BiFunction, Stream, ...
5/12/25	Review	In-class Q/A to prepare for final

Finals	Final Exam	Two problems, all of the finals period. Monday, May 19 8:30-10:30 AM, MH 422
--------	------------	--