

# Applied Algorithms Section 01 CS 48

Spring 2024 1 Unit(s) 01/24/2024 to 05/13/2024 Modified 02/09/2024

## Contact Information

Instructor: Dr. David Scot Taylor

Email: [david.taylor@sjsu.edu](mailto:david.taylor@sjsu.edu)

Office: MacQuarrie Hall 212

Phone: 4-5124

Website: [www.cs.sjsu.edu/~taylor](http://www.cs.sjsu.edu/~taylor)

### Office Hours

Monday, 1:30 PM to 2:30 PM, MacQuarrie Hall 212

Tuesday, 10:30 AM to 11:30 AM, MacQuarrie Hall 212

Also by appointment, or over zoom if required: <https://sjsu.zoom.us/j/89560963006>  
(<https://sjsu.zoom.us/j/89560963006>)

## Course Information

### Lecture

Tuesday, 12:00 PM to 12:50 PM, Interdisciplinary Science 782

## 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 will 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

This is your class. Please ask questions. Please come prepared. Do not engage in activity that may distract other students.

I do not take attendance except to establish enrollment for the first few classes, and of course for exams. Students not attending either of the first two classes will be dropped. Attempting to get marked as present (by having someone else attend in your place or using technological deceptions) will be considered academic dishonesty and at a minimum will result in you getting dropped from the course.

## Program Information

---

Diversity Statement - At SJSU, it is important to create a safe learning environment where we can explore, learn, and grow together. We strive to build a diverse, equitable, inclusive culture that values, encourages, and supports students from all backgrounds and experiences.

## Course Goals

---

This course intends to help you to apply the concepts you have learned in CS 46B. We do this by writing lots of code--every week we will pick a new problem to solve. You will work on the solution BY YOURSELF. The next class we will go over the solution to the problem and start a new one. The key is that you practice your ability to write code to apply concepts that you have learned and thereby develop confidence in your programming ability.

## Course Learning Outcomes (CLOs)

---

Upon successful completion of this course, students will be able to:

1. demonstrate proper usage of APIs for data input and output.
2. evaluate when basic data structures such as arrays, linked lists, and hash tables should be used.
3. develop solutions to common programming problems used in industry to test programming ability.
4. explain solutions to problems and implement them in code.
5. analyze potential performance problems in code and devise solutions to improve performance.
6. implement an abstract description of a solution in code.

## Course Materials

---

### Computer

Students should have a laptop available, with Java and an IDE installed.

## Course Requirements and Assignments

---

## 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 video (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 presentation (3-5 mins, either in-person or a video, TBD) to show an awesome feature of your favorite IDE. I will schedule these and set up topics. (It makes sense to schedule more basic topics earlier in the semester.)

## Program Explanation (5%)

At some point during the semester, each student will be expected to explain a solution to a programming problem to the class, or to the instructor individually.

## Final Examination and in class programming problem (30% total)

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

## Grading Information

---

This is a graded class, but I will use minimum grading: you cannot get below a 50% on any submission or exam. For example, if you do not submit a solution explanation or your submission falls far short and only scores 35%, you will be assigned a 50% in the grade book. The minimum grading does not apply to cases of academic integrity.

programming assignments	55%
IDE tip video	10%
solution explanation	5%
in class programming problems	30%

Your course grade will be determined by your final weighted average:

*A plus = 97% or higher*

*A = 93% up to 97%*

*A minus = 90% to 93%*

*B plus = 87% to 90%*

*B = 83% to 87%*

*B minus = 80% to 83%*

*C plus = 77% to 80%*

*C = 73% to 77%*

*C minus = 70% to 73%*

*D plus = 67% to 70%*

*D = 63% to 67%*

*D minus = 60% to 63%*

*F = 0% to 60%*

There is no rounding, but boundary cases count as the higher of the two grades. I reserve the option to make cutoffs more lenient, but not more strict.

Per [University Policy S16-9 \(PDF\)](http://www.sjsu.edu/senate/docs/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) (<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

---

The following is a rough course schedule

<b>Week/Lesson/Module</b>	<b>Topics, Readings, Assignments, Deadlines</b>
1	Input/Output
2	Input/Output
3	Strings
4	Math/bit operations
5	Arrays
6	Arrays
7	Arrays/searching
8	Arrays/sorting
9	In-class problem (in-person attendance required) 3/26/2024
10	Linked Lists
11	Hashtables
12	Hashtables

Week/Lesson/Module	Topics, Readings, Assignments, Deadlines
13	Simple big O
14	Bit vectors
15	Lambdas?
16	Performance Issues?
Final	In-class problem (Monday 5/20/2024 @ 9:45am-noon)