

# Unit 4: Standard Collection Algorithms

Content Area: **Computer Science**  
Course(s):  
Time Period: **Marking Period 1**  
Length: **2-3 Weeks**  
Status: **Published**

## Summary

---

|                        |   |
|------------------------|---|
| CS.9-12.8.1.12.AP.1    | Design algorithms to solve computational problems using a combination of original and existing algorithms.  |
| CS.9-12.8.1.12.AP.3    | Select and combine control structures for a specific application based upon performance and readability, and identify trade-offs to justify the choice.   |
| CS.9-12.8.1.12.AP.4    | Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.   |
| LA.K-12.NJSLSA.L4      | Determine or clarify the meaning of unknown and multiple-meaning words and phrases by using context clues, analyzing meaningful word parts, and consulting general and specialized reference materials, as appropriate. |
| LA.K-12.NJSLSA.L5      | Demonstrate understanding of word relationships and nuances in word meanings.   |
| MA.9-12.1.2.12prof.Cr  | Creating  |
| MA.9-12.1.2.12prof.Cr2 | Organizing and developing ideas.  |
| TECH.8.2.12.E.1        | Demonstrate an understanding of the problem-solving capacity of computers in our world.   |
| TECH.8.2.12.E.2        | Analyze the relationships between internal and external computer components.  |
| TECH.8.2.12.E.3        | Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications, and games).  |
| TECH.8.2.12.E.4        | Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements).  |

## Essential Questions / Enduring Understandings

---

Essential Questions:

- What algorithm would you choose to solve a given problem? Why?
- How do you traverse an array?
- How are elements inserted into an already established array?
- How are elements deleted from an already established array?
- What is an abstract data type?
- How do you search an array?
- Which search method is more efficient?
- How do you sort an array?
- Which sort method is best?
- Which search method is more efficient?

Enduring Understandings:

- Arrays can be manipulated.
- Different search algorithms exist and when best to use each.
- Different sort algorithms exist and when best to use each.

## Objectives

---

Students will know:

- operations on arrays.
- how to traverse an array
- how to insert elements into an array.
- how to delete elements from an array.
- how to sort an array using either bubble, selection, insertion, or merge sort algorithms.
- how to perform a sequential search.
- how to perform a binary search.

Students will be skilled at:

- recognizing sorting algorithms.
- iterating through a collection of data.

## Learning Plan

---

- Preview the essential questions and connect to learning throughout the unit.
- Algorithm for accessing arrays
- Traversing arrays.
- Insertion and deletion from an array
- Selection and insertion of sorts.
- Merge sort (recursion - optional)
- Discussion of efficiency of the three different sorts.
- Searches (sequential and binary)
- Identify boundary cases for programs and how best to test those boundary cases.
- Discuss the use of pre- and post-conditions when writing programs. Using already written programs, have students identify

the pre- and post-conditions.

## Assessment

---

- Assessments
  - Formative: Daily assessments using examples from class notes and CodeHS.com, AP Classroom/Albert Checks for Understanding
  - Summative: Teacher-created assessments/projects and CodeHS Computer Science Projects, AP Classroom/Albert Unit Assessments
  - Benchmark: Check for understanding benchmark assessments on CodeHS, AP Classroom/Albert/Khan Academy Diagnostics
  - Alternative Assessments: Student-centered activities such as a doorbell coding project, game design projects, and other activities involving real world applications
- Complete performance tasks: be able to design programs using appropriate code; be able to write programs using appropriate code.
- Complete quizzes/test: Operations on data structures, Searching/Sorting
- Teacher observation of students doing work on the performance tasks.
- Conduct self-assessments and reflections
- Conduct peer evaluations
- Participate in class discussions
- Sample AP multiple choice questions

## Materials

---

- Core instructional materials: [Core Book List](#)

Supplemental materials: CodeHS

- Computer
- Reference books

## Integrated Accommodation and Modification

---

See [Linked Document](#).

