Unit 07: Array Lists

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

Summary

As students learned in Unit 6, data structures are helpful when storing multiple related data values. Arrays have a static size, which causes limitations related to the number of elements stored, and it can be challenging to reorder elements stored in arrays. The ArrayList object has a dynamic size, and the class contains methods for insertion and deletion of elements, making reordering and shifting items easier. Deciding which data structure to select becomes increasingly important as the size of the data set grows, such as when using a large real-world data set.

In this unit, students will also learn about privacy concerns related to storing large amounts of personal data and about what can happen if such information is compromised.

Revision Date: July 2026

L.SS.11–12.1	Demonstrate command of the system and structure of the English language when writing or speaking.
P.3	Consider the environmental, social and economic impacts of decisions.
P.5	Utilize critical thinking to make sense of problems and persevere in solving them.
P.5	Utilize critical thinking to make sense of problems and persevere in solving them.
P.6	Model integrity, ethical leadership and effective management.
L.KL.11–12.2	Apply knowledge of language to understand how language functions in different contexts, to make effective choices for meaning or style, and to comprehend more fully when reading or listening.
P.8	Use technology to enhance productivity increase collaboration and communicate effectively.
P.9	Work productively in teams while using cultural/global competence.
8.1.12.DA.2	Describe the trade-offs in how and where data is organized and stored.
8.1.12.AP.1	Design algorithms to solve computational problems using a combination of original and existing algorithms.
9.4.12.IML.3	Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions (e.g., S-ID.B.6a., 8.1.12.DA.5, 7.1.IH.IPRET.8).
8.1.12.AP.2	Create generalized computational solutions using collections instead of repeatedly using simple variables.
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.
8.1.12.AP.5	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
9.4.12.IML.7	Develop an argument to support a claim regarding a current workplace or societal/ethical issue such as climate change (e.g., NJSLSA.W1, 7.1.AL.PRSNT.4).

8.1.12.AP.6	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.

Essential Questions & Essential Understanding

- How are algorithms different for arrays and ArrayLists?
- How do you traverse an ArrayList?
- How is data collection viewed ethically?
- What is an ArrayList?
- What is auto-boxing and auto-unboxing?
- What methods exist for ArrayLists?
- What searching algorithms can be used in ArrayLists?
- What sorting algorithms are most efficient for various forms of data?

Objectives

Students Will Know

- how to create and access data in an ArrayList.
- how to minipulate data using ArrayList methods.
- the difference between array and ArrayList data analysis.
- the sorting algorithms of selection, insertion, and bubble. (Merge Sort is in Unit 10).
- the searching algorithms of linear and binary.

Students Will Be Skilled At

- manipulating data in a collection in order to simulate a situation.
- analyzing data within an ArrayList.
- searching and sorting data within an ArrayList.

Learning Plan

Lecture and demonstration on ArrayLists and methods within.

Discussion on auto-boxing and auto-unboxing.

Pair programming exercises on ArrayList methods and common errors.

Individual lab on searching for data in given datasets. Students start with either sorted data or unsorted data and create algorithms (sequential - unsorted, binary - unsorted) for finding the appropriate value. Efficiency is looked at and big O notation introduced but not tested.

Lecture and classroom activity on sorting algorithms.

Assessments

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:

- Students will be able to design programs using ArrayLists to hold data.
- Students will be able to write programs solving problems using ArrayLists.

complete quizzes/tests:

- ArrayList creation and methods.
- Common errors surrounding ArrayLists
- Iterating through datasets
- searching algorithms
- sorting algorithms

complete sample AP multiple choice questions.

complete sample AP open ended questions.

Materials

Core instructional materials: Computing Concepts with Java

Supplemental instructional materials:

Java Concepts for AP Computer Science Study Guide
CollegeBoard AP Classroom Website
CollegeBoard AP Computer Science A Website

Integrated Accomodations & Modifications

Possible accommodations/modification for AP Computer Science A