

AP COMPUTER SCIENCE A

UNIT 4

Iteration



17.5–22.5%
AP EXAM WEIGHTING



~14–16
CLASS PERIODS

The icon consists of a white circle containing a blue square with the letters 'AP' in white. Below the square is a small blue monitor-like shape with two vertical lines representing a stand.

Remember to go to [AP Classroom](#) to assign students the online **Personal Progress Check** for this unit.

Whether assigned as homework or completed in class, the **Personal Progress Check** provides each student with immediate feedback related to this unit's topics and skills.

Personal Progress Check 4

Multiple-choice: ~15 questions

Free-response: 2 questions

- Methods and Control Structures
- Methods and Control Structures: partial

Iteration



Developing Understanding

BIG IDEA 1

Control **CON**

- How does iteration improve programs and reduce the amount of program code necessary to complete a task?
- What situations would warrant the use of one type of loop over another?

This unit focuses on iteration using `while` and `for` loops. As you saw in Unit 3, Boolean expressions are useful when a program needs to perform different operations under different conditions. Boolean expressions are also one of the main components in iteration. This unit introduces several standard algorithms that use iteration. Knowledge of standard algorithms makes solving similar problems easier, as algorithms can be modified or combined to suit new situations.

Iteration is used when traversing data structures such as arrays, ArrayLists, and 2D arrays. In addition, it is a necessary component of several standard algorithms, including searching and sorting, which will be covered in later units.

Building Computational Thinking Practices

2.B 2.D 3.C 5.C

Students should be able to determine the result of program code that uses iterative statements to represent nonlinear processes in a program. Students should practice determining the number of times a given loop structure will execute. Spending time analyzing existing program code provides opportunities for students to better understand how iterative structures can be set up and used to solve their own problems, as well as the implications associated with code changes, such as how using a different iterative structure might change the result of a set of program code.

Students should be able to implement program code that uses iterative statements to represent nonlinear processes. Understanding how to write program code that repeats allows students to write programs to solve a wider variety of problems, including those that use data, which will be covered in Units 6, 7, and 8.

Preparing for the AP Exam

While the concept of iteration is tested in isolation on the multiple-choice exam, it is a foundational concept that students will use along with other topics, such as data structures, on free-response questions. Often, students struggle in situations that warrant variation in the Boolean condition of loops, such as when they want to terminate a loop early. Early termination of a loop requires the use of conditional statements within the body of the loop. If the order of the program statements is incorrect, the early termination may be triggered too early or not at all. Provide students with practice ordering statements by giving them strips of paper, each with a line of program code that can be used to assemble the correct and incorrect solutions. Ask them to reassemble the code and trace it to see if it is correct. Using manipulatives in this way makes it easier for students to rearrange the order of the program code to determine if it is in the correct order.

UNIT AT A GLANCE

Enduring Understanding	Topic	Suggested Skills	Class Periods
			~14–16 CLASS PERIODS
CON-2	4.1 while Loops	<p>1.B Determine code that would be used to complete code segments.</p> <p>2.B Determine the result or output based on statement execution order in a code segment without method calls (other than output).</p> <p>3.C Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.</p>	
	4.2 for Loops	<p>3.C Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.</p> <p>4.C Determine if two or more code segments yield equivalent results.</p> <p>5.C Explain how the result of program code changes, given a change to the initial code.</p>	
	4.3 Developing Algorithms Using Strings	<p>2.C Determine the result or output based on the statement execution order in a code segment containing method calls.</p> <p>3.C Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.</p>	
	4.4 Nested Iteration	<p>1.B Determine code that would be used to complete code segments.</p> <p>3.C Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.</p> <p>5.C Explain how the result of program code changes, given a change to the initial code.</p>	
	4.5 Informal Code Analysis	<p>2.D Determine the number of times a code segment will execute.</p>	
<p>Go to AP Classroom to assign the Personal Progress Check for Unit 4. Review the results in class to identify and address any student misunderstandings.</p>			

SAMPLE INSTRUCTIONAL ACTIVITIES

The sample activities on this page are optional and are offered to provide possible ways to incorporate instructional approaches into the classroom. They were developed in partnership with teachers from the AP community to share ways that they approach teaching some of the topics in this unit. Please refer to the Instructional Approaches section beginning on p. 159 for more examples of activities and strategies.

Activity	Topic	Sample Activity
1	4.2	<p>Jigsaw</p> <p>As a whole class, look at a code segment containing iteration and the resulting output. Afterward, divide students into groups, and provide each group with a slightly modified code segment. After groups have determined how the result changes based on their modified segment, have them get together with students who investigated a different version of the code segment and share their conclusions.</p>
2	4.1–4.4	<p>Note-taking</p> <p>Provide students with a method that, when given an integer, returns the month name from a <code>String</code> that includes all the month names in order, each separated by a space. Have them annotate what each statement does in the method. Then, ask students to use their annotated method as a guide to write a similar method that, given a student number as input, returns the name of a student from a <code>String</code> containing the first name of all students in the class, each separated by a space.</p>
3	4.5	<p>Simplify the problem</p> <p>Provide students with several code segments containing iteration. For each segment, have students trace through the execution of a loop with smaller bounds to see what boundary cases are considered, and then use that information to determine the number of times each loop executes with the original bounds.</p>



Unit Planning Notes

Use the space below to plan your approach to the unit. Consider how you want to pace your course and where you will incorporate writing and analyzing program code.

.....

.....

.....



After completing this unit, students will have covered all of the necessary content for the Consumer Review Lab. The proposed class periods for this unit include time to complete the provided lab activities.

SUGGESTED SKILLS

1.B

Determine code that would be used to complete code segments.

2.B

Determine the result or output based on statement execution order in a code segment without method calls (other than output).

3.C

Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.



AVAILABLE RESOURCES

- [Runestone Academy: AP CSA—Java Review: 7.2—While Loops](#)
- [Practice-It!: BJP4 Chapter 5: Program Logic and Indefinite Loops—Exercises 5.1–5.4](#)
- The Exam > [2017 AP Computer Science A Exam Free-Response Question #3, Part B \(Phrase\)](#)

TOPIC 4.1

while Loops

Required Course Content

ENDURING UNDERSTANDING

CON-2

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

LEARNING OBJECTIVE

CON-2.C

Represent iterative processes using a `while` loop.

ESSENTIAL KNOWLEDGE

CON-2.C.1

Iteration statements change the flow of control by repeating a set of statements zero or more times until a condition is met.

CON-2.C.2

In loops, the Boolean expression is evaluated before each iteration of the loop body, including the first. When the expression evaluates to `true`, the loop body is executed. This continues until the expression evaluates to `false`, whereupon the iteration ceases.

CON-2.C.3

A loop is an infinite loop when the Boolean expression always evaluates to `true`.

CON-2.C.4

If the Boolean expression evaluates to `false` initially, the loop body is not executed at all.

CON-2.C.5

Executing a `return` statement inside an iteration statement will halt the loop and exit the method or constructor.

continued on next page

LEARNING OBJECTIVE

CON-2.D

For algorithms in the context of a particular specification that does not require the use of traversals:

- Identify standard algorithms.
- Modify standard algorithms.
- Develop an algorithm.

ESSENTIAL KNOWLEDGE

CON-2.D.1

There are standard algorithms to:

- Identify if an integer is or is not evenly divisible by another integer
- Identify the individual digits in an integer
- Determine the frequency with which a specific criterion is met

CON-2.D.2

There are standard algorithms to:

- Determine a minimum or maximum value
- Compute a sum, average, or mode

SUGGESTED SKILLS

3.C

Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.

4.C

Determine if two or more code segments yield equivalent results.

5.C

Explain how the result of program code changes, given a change to the initial code.



AVAILABLE RESOURCES

- [Runestone Academy: AP CSA—Java Review: 7.3—For Loops](#)
- [Practice-It!: BJP4 Chapter 2: Primitive Data and Definite Loops—Exercises 2.2, 2.3](#)

TOPIC 4.2

for Loops

Required Course Content

ENDURING UNDERSTANDING

CON-2

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

LEARNING OBJECTIVE

CON-2.E

Represent iterative processes using a `for` loop.

ESSENTIAL KNOWLEDGE

CON-2.E.1

There are three parts in a `for` loop header: the initialization, the Boolean expression, and the increment. The increment statement can also be a decrement statement.

CON-2.E.2

In a `for` loop, the initialization statement is only executed once before the first Boolean expression evaluation. The variable being initialized is referred to as a loop control variable.

CON-2.E.3

In each iteration of a `for` loop, the increment statement is executed after the entire loop body is executed and before the Boolean expression is evaluated again.

CON-2.E.4

A `for` loop can be rewritten into an equivalent `while` loop and vice versa.

CON-2.E.5

“Off by one” errors occur when the iteration statement loops one time too many or one time too few.

TOPIC 4.3

Developing Algorithms Using Strings

Required Course Content

ENDURING UNDERSTANDING

CON-2

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

LEARNING OBJECTIVE

CON-2.F

For algorithms in the context of a particular specification that involves `String` objects:

- Identify standard algorithms.
- Modify standard algorithms.
- Develop an algorithm.

ESSENTIAL KNOWLEDGE

CON-2.F.1

There are standard algorithms that utilize `String` traversals to:

- Find if one or more substrings has a particular property
- Determine the number of substrings that meet specific criteria
- Create a new string with the characters reversed

SUGGESTED SKILLS**2.C**

Determine the result or output based on the statement execution order in a code segment containing method calls.

3.C

Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.

**AVAILABLE RESOURCES**

- **Practice-It!:** [BJP4 Chapter 3: Parameters and Objects—Exercise 3.19](#)
- **Practice-It!:** [BJP4 Chapter 5: Program Logic and Indefinite Loops—Exercise 5.24](#)
- **CodingBat Java:** [String-2](#)

SUGGESTED SKILLS

1.B

Determine code that would be used to complete code segments.

3.C

Write program code to satisfy method specifications using expressions, conditional statements, and iterative statements.

5.C

Explain how the result of program code changes, given a change to the initial code.



AVAILABLE LAB

- Classroom Resources > [AP Computer Science A: Consumer Review Lab](#)

AVAILABLE RESOURCES

- [Runestone Academy: AP CSA—Java Review: 7.4—Nested For Loops](#)
- [Practice-It!: BJP4 Chapter 2: Primitive Data and Definite Loops—Exercises 2.4–2.15](#)
- Past AP Free-Response Exam Questions on Methods and Control Structures on AP Question Bank

TOPIC 4.4

Nested Iteration

Required Course Content

ENDURING UNDERSTANDING

CON-2

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

LEARNING OBJECTIVE

CON-2.G

Represent nested iterative processes.

ESSENTIAL KNOWLEDGE

CON-2.G.1

Nested iteration statements are iteration statements that appear in the body of another iteration statement.

CON-2.G.2

When a loop is nested inside another loop, the inner loop must complete all its iterations before the outer loop can continue.

TOPIC 4.5

Informal Code Analysis

SUGGESTED SKILL**2.D**

Determine the number of times a code segment will execute.

Required Course Content

ENDURING UNDERSTANDING

CON-2

Programmers incorporate iteration and selection into code as a way of providing instructions for the computer to process each of the many possible input values.

LEARNING OBJECTIVE

CON-2.H

Compute statement execution counts and informal run-time comparison of iterative statements.

ESSENTIAL KNOWLEDGE

CON-2.H.1

A statement execution count indicates the number of times a statement is executed by the program.

