

CompuScholar, Inc.
Alignment to Utah
"AP Computer Science A" Course Standards

Utah Course Details:

Course Name:	Advanced Placement Computer Science A
Primary Cluster:	CTE / IT
Course Code(s):	35.02.00.00.041
Credit:	1.0
Grade Level:	10th-12th
State Standards Link:	AP Computer Science Strands and Standards (June 2016)

CompuScholar Course Details:

Course Title:	Java Programming (AP)
Course ISBN:	978-0-9887070-2-3
Course Year:	2019

Note 1: Citation(s) listed may represent a subset of the instances where objectives are met throughout the course.

Note 2: Citation(s) for a "Lesson" refer to the "Lesson Text" elements and associated "Activities" within the course, unless otherwise noted. The "Instructional Video" components are supplements designed to introduce or re-enforce the main lesson concepts, and the Lesson Text contains full details.

Course Description

AP Computer Science A emphasizes object-oriented programming methodology with an emphasis on problem solving and algorithm development and is meant to be the equivalent of a first-semester course in computer science. It also includes the study of data structures and abstraction.

CompuScholar's Java Programming (AP) course is on the College Board's list of example textbooks. A pre-approved Course Audit syllabus and a topic cross-reference table are posted on the course description page.

<https://www.compuscholar.com/schools/courses/ap-java/>

Course Standards

STRAND 1: Students will use Object-Oriented Program Design	CITATION(S)
Standard 1: Program and Class Design	
a. Problem analysis	Chapter 15, Lessons 2 - 6 Chapter 17, Lesson 4 Chapter 20, Lesson 1 Chapter 22, Lessons 1-2
b. Data abstraction and encapsulation	Chapter 10, Lesson 1

c. Class specifications, interface specifications, relationships (“is-a,” “has-a”), and extension using inheritance	Chapter 10, Lessons 1 - 2 Chapter 11, Lesson 2 Chapter 15, Lessons 2 - 6
d. Code reuse	Chapters 8, 10, 11, 15
e. Data representation and algorithms	Chapter 10, Lesson 2 Chapter 17, Lesson 4 Chapter 20, Lesson 1
f. Functional decomposition	Chapter 8, Lesson 1 Chapter 22, Lesson 1

STRAND 2: Students will use Program Implementation	CITATION(S)
Standard 1: Implementation techniques	
a. Top-down	Chapter 24, Lesson 1
b. Bottom-up	Chapter 24, Lesson 1
c. Object-oriented	Chapters 10, 11, 15
d. Encapsulation and information hiding	Chapter 10, Lessons 1, 3
e. Procedural abstraction	Chapter 8 Chapter 10, Lesson 1 Chapter 15, Lessons 2 - 4
Standard 2: Programming constructs	
a. Primitive types vs. reference types	Chapter 4, Lessons 1 - 2 Chapter 5, Lesson 1
b. Declaration	Chapter 4, Lessons 1 - 2 Chapter 5, Lesson 1
c. Constants	Chapter 4, Lesson 2
d. Variables	Chapter 4, Lesson 2
e. Methods and parameters	Chapter 8
f. Classes	Chapters 10, 11, 15
g. Interfaces	Chapter 11, Lesson 2
h. Text output using System.out.print and System.out.println	Chapter 2, Lesson 2 Chapter 4, Lesson 3

i. Control	Chapter 7, Lessons 2 - 3
j. Method call	Chapter 8
k. Sequential execution	Chapter 2, Lesson 2
l. Conditional execution	Chapter 7, Lessons 2 - 3
m. Iteration	Chapter 7, Lessons 4 - 5
n. Recursion	Chapter 19, Lesson 1
o. Expression evaluation	Chapter 7, Lesson 1
p. Numeric expressions	Chapter 4, Lesson 2 Chapter 7, Lesson 1
q. String expressions	Chapter 5
r. Boolean expressions, short-circuit evaluation, De Morgan's law	Chapter 7, Lesson 1
Standard 3: Java library classes and interfaces included in the AP Java Subset	Throughout the course (See cross-reference document).

STRAND 3: Students will use Program Analysis.	CITATION(S)
Standard 1: Testing	
a. Development of appropriate test cases, including boundary cases	Chapter 9, Lesson 3 Chapter 24, Lesson 3 Chapter 25, Lesson 4
b. Unit testing	Chapter 9, Lesson 3 Chapter 24, Lesson 3 Chapter 25, Lesson 4
c. Integration testing	Chapter 9, Lesson 3 Chapter 24, Lesson 3 Chapter 25, Lesson 4
Standard 2: Debugging	
a. Error categories: compile-time, run-time, logic	Chapter 9, Lesson 1
b. Error identification and correction	Chapter 9, Lessons 1, 3
c. Techniques such as using a debugger, adding extra output statements, or hand-tracing code.	Chapter 9, Lessons 3 - 4

Standard 3: Runtime exceptions	Chapter 9, Lesson 1
Standard 4: Program correctness	
a. Pre- and post-conditions	Chapter 24, Lesson 3
b. Assertions	Chapter 24, Lesson 3
Standard 5: Algorithm Analysis	
a. Statement execution counts	Chapter 20, Lessons 2 - 3
b. Informal running time comparison	Chapter 20, Lessons 2 - 3
Standard 6: Numerical representations of integers	
a. Representations of non-negative integers in different bases	Chapter 17, Lesson 2
b. Implications of finite integer bounds	Chapter 17, Lesson 2

STRAND 4: Students will use Standard Data Structures	CITATION(S)
Standard 1: Primitive data types (int, boolean, double)	Chapter 4, Lessons 1 - 2
Standard 2: Strings	Chapter 5
Standard 3: Classes	Chapters 10, 11, 15
Standard 4: Lists	Chapter 14, Lessons 3 - 4
Standard 5: Arrays (1-dimensional and 2-dimensional)	Chapter 14, Lessons 1 - 2

STRAND 5: Students will use Standard Operations and Algorithms	CITATION(S)
Standard 1: Operations on data structures	
a. Traversals	Chapter 14, Lesson 5
b. Insertions	Chapter 14, Lessons 3 - 4
c. Deletions	Chapter 14, Lessons 3 - 4

Standard 2: Searching	
a. Sequential	Chapter 19, Lesson 3
b. Binary	Chapter 19, Lesson 3

STRAND 6: Students will use Computing in Context	CITATION(S)
Standard 1: System reliability	Chapter 1, Lesson 4 Chapter 24, Lesson 3 Suppl. Chapter 2, Lesson 1
Standard 2: Privacy	Chapter 1, Lesson 4 Suppl. Chapter 3, Lesson 1
Standard 3: Legal issues and intellectual property	Chapter 1, Lesson 4
Standard 4: Social and ethical ramifications of computer use	Chapter 1, Lesson 4 Suppl. Chapter 3, Lesson 2