



CS Standards Crosswalk: CSTA K-12 Computer Science Standards and Oracle Java Fundamentals (2014)

CSTA Website	http://csta.acm.org/Curriculum/sub/K12Standards.html
Oracle Website	https://academy.oracle.com/oa-web-introcs-curriculum.html
Oracle Contact	academy_ww@oracle.com

Level 2 (recommended for grades 6–9) Computer Science and Community

Middle school/junior high school students begin using computational thinking as a problem-solving tool. They begin to appreciate the ubiquity of computing and the ways in which computer science facilitates communication and collaboration. Students begin to experience computational thinking as a means of addressing issues relevant, not just to them, but to the world around them.

(● = terminal objective: meets all or part of CSTA standard; ○ = enabling objective: an early step that could lead to, but is not sufficient for, meeting CSTA standard)

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computational Thinking
Use the basic steps in algorithmic problem-solving to design solutions (e.g., problem statement and exploration, examination of sample instances, design, implementing a solution, testing and evaluation).	<ul style="list-style-type: none"> ● Code a simple programming instruction [2.1.5] ● Test and debug an animation [2.3.8] ● Write a storyboard [2.5.2] ● Flowchart a storyboard [2.5.3] ● Complete an animation [2.12.2] ● Test an animation [2.12.3] ● Demonstrate program testing strategies [3.4.1] ● List programming tasks to design and implement a game using Java [3.12.4] ● Compile an application [4.1.3] ● Test to ensure application is complete [4.1.4] ● Modify a program to execute error free [4.1.6] ○ Understand the value of testing and debugging [2.1.7] ○ Recognize phases for developing a software application [3.4.2] ○ Identify components of a Java application [4.1.2] 	<ul style="list-style-type: none"> ● Get Started with Alice 3 [2.1] ● Use Procedures and Arguments [2.3] ● Declare Procedures [2.5] ● Develop a Complete Animation [2.12] ● Developing and Testing an Application [3.4] ● Creating an Inventory of Java Fundamentals [3.12] ● Compiling with Eclipse – A First Program [4.1]
Describe the process of parallelization as it relates to problem solving.		
Define an algorithm as a sequence of instructions that can be processed by a computer.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computational Thinking
Evaluate ways that different algorithms may be used to solve the same problem.	<ul style="list-style-type: none"> • Compare and contrast search and sort algorithms [6.2.5] • Compare the pros and cons of recursion [7.3.9] 	<ul style="list-style-type: none"> • Sorting and Searching [6.2] • Understanding Recursion, the Static Modifier, and Nested Classes [7.3]
Act out searching and sorting algorithms.		
Describe and analyze a sequence of instructions being followed (e.g., describe a character's behavior in a video game as driven by rules and algorithms).		
Represent data in a variety of ways including text, sounds, pictures and numbers.		
Use visual representations of problem states, structures and data (e.g., graphs, charts, network diagrams, flowcharts).	<ul style="list-style-type: none"> • Correlate storyboard statements with program execution tasks [2.4.1] • Write a storyboard [2.5.2] • Flowchart a storyboard [2.5.3] • Demonstrate and explain UML (Unified Modeling Language) class diagrams [7.4.1] 	<ul style="list-style-type: none"> • Add Rotation and Randomization [2.4] • Declare Procedures [2.5] • Understanding Inheritance [7.4]
Interact with content-specific models and simulations (e.g., ecosystems, epidemics, molecular dynamics) to support learning and research.		
Evaluate what kinds of problems can be solved using modeling and simulation.		
Analyze the degree to which a computer model accurately represents the real world.		
Use abstraction to decompose a problem into sub problems.	<ul style="list-style-type: none"> • Identify and use procedural abstraction techniques to simplify animation development [2.5.7] • Use functional decomposition to write a scenario and storyboard [2.12.1] • Define abstraction and provide an example of when it is used [3.9.1] ○ Describe the difference between an Object class and a Driver class [4.2.2] 	<ul style="list-style-type: none"> • Declare Procedures [2.5] • Develop a Complete Animation [2.12] • Understanding Abstraction [3.9] • Using Object Classes and Driver Classes [4.2]

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computational Thinking
Understand the notion of hierarchy and abstraction in computing including high level languages, translation, instruction set and logic circuits.	<ul style="list-style-type: none"> • Create a multilevel class hierarchy [7.4.7] • Recognize when constructors are called in a class hierarchy [7.4.8] 	<ul style="list-style-type: none"> • Understanding Inheritance [7.4]
Examine connections between elements of mathematics and computer science including binary numbers, logic, sets and functions.	<ul style="list-style-type: none"> • Create an expression using logic operators [3.10.4] <i>[CSTA standard is partially met by the listed objective.]</i> 	<ul style="list-style-type: none"> • Using Loops, Variables, and Strings [3.10]
Provide examples of interdisciplinary applications of computational thinking.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Collaboration
Apply productivity/ multimedia tools and peripherals to group collaboration and support learning throughout the curriculum.		
Collaboratively design, develop, publish and present products (e.g., videos, podcasts, websites) using technology resources that demonstrate and communicate curriculum concepts.	<ul style="list-style-type: none"> ○ Plan the presentation of a completed animation project [2.12.5] 	<ul style="list-style-type: none"> • Develop a Complete Animation [2.12]
Collaborate with peers, experts and others using collaborative practices such as pair programming, working in project teams and participating in-group active learning activities.	<ul style="list-style-type: none"> ○ Describe the importance of teamwork and introduce a member of the class [1.1.6] ○ Describe the components of a team project [1.2.3] ○ Create a teamwork assessment rubric [1.2.4] 	<ul style="list-style-type: none"> • Welcome [1.1] • Introduction [1.2]
Exhibit dispositions necessary for collaboration: providing useful feedback, integrating feedback, understanding and accepting multiple perspectives, socialization.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9) Strand: Computing Practice and Programming		
Select appropriate tools and technology resources to accomplish a variety of tasks and solve problems.	<ul style="list-style-type: none"> • Access a minimum of two Java class APIs [4.2.3] • Use a method from the Math class [4.3.9] • Access a Math class method from the Java API [4.3.10] 	<ul style="list-style-type: none"> • Using Object Classes and Driver Classes [4.2] • Programming with Data Types and Operators [4.3]
Use a variety of multimedia tools and peripherals to support personal productivity and learning throughout the curriculum.		
Design, develop, publish and present products (e.g., web pages, mobile applications, animations) using technology resources that demonstrate and communicate curriculum concepts.	<ul style="list-style-type: none"> ○ Plan the presentation of a completed animation project [2.12.5] 	<ul style="list-style-type: none"> • Develop a Complete Animation [2.12]
Demonstrate an understanding of algorithms and their practical application.		
Implement problem solutions using a programming language, including: looping behavior, conditional statements, logic, expressions, variables and functions.	<ul style="list-style-type: none"> • Demonstrate how to declare a procedure [2.5.6] • Use the IF control structure to effect execution of instructions [2.8.1] • Use the WHILE control structure to create a conditional loop for repetitive behavior [2.8.2] • Create an instance of a class [3.1.3] • Demonstrate source code changes to invoke methods programmatically [3.3.1] • Demonstrate source code changes to write an IF decision statement [3.3.2] • Create IF-ELSE control statements [3.5.3] • Create an instance of a class [3.5.4] • Simplify programming by creating and calling defined methods [3.6.2] • Construct a world object using a constructor method [3.8.1] • Create an object using a constructor [3.8.2] • Write programming statements to use the new keyword [3.8.3] • Write programming statements to switch between two images [3.8.6] • Write programming statements to end a game [3.8.7] • Create a while loop in a constructor to build a world [3.10.1] • Create an expression using logic operators [3.10.4] • Use string variables to store and concatenate strings [3.10.6] 	<ul style="list-style-type: none"> • Declare Procedures [2.5] • Use the IF and WHILE Control Structures [2.8] • Use Variables [2.10] • Correlating Java Variables, Data Types, and Expressions with Alice 3 Tools [2.13] • Correlating Java Methods, Classes, and Other Structures with Alice 3 Tools [2.14] • Getting Started With Greenfoot [3.1] • Using Methods, Variables and Parameters [3.2] • Working with Source Code and Documentation [3.3] • Using Randomization and Understanding Dot Notation and Constructors [3.5] • Defining Methods [3.6] • Creating a World, Animating Actors, and Ending a Game [3.8] • Using Loops, Variables, and Strings [3.10] • Putting it All Together with Greenfoot [3.11] • Using Object Classes and Driver Classes [4.2]

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)	Strand: Computing Practice and Programming	
	<ul style="list-style-type: none"> • Apply Greenfoot knowledge by creating a Java game [3.11.1] • Create an Object class [4.2.5] • Create a Driver class [4.2.6] • Use primitive data types in Java code [4.3.1] • Specify literals for the primitive types and for Strings [4.3.2] • Demonstrate how to initialize variables [4.3.3] • Apply casting in Java code [4.3.6] • Use arithmetic operators [4.3.7] • Use the assignment operator [4.3.8] • Instantiate (create) a String [4.4.1] • Use the + and += operators for concatenating Strings [4.4.3] • Test Strings with the compareTo() and equals() method [4.4.6] • Use String methods length(), substring(), indexOf(), and charAt() [4.4.8] • Use if-else logic and statements [5.1.2] • Apply switch logic and statements in Java code [5.1.3] • Use break and default effectively in a switch statement [5.1.4] • Use the ternary operator [5.1.5] • Create a while loop [5.2.1] • Create a do-while loop [5.2.2] • Create a for loop [5.2.3] • Create an object of a class [7.1.2] • Create methods that compile with no errors [7.1.3] • Return a value from a method [7.1.4] • Use parameters in a method [7.1.5] • Create a driver class and add instances of Object classes [7.1.6] • Add a constructor to a class [7.1.7] • Apply the new operator [7.1.8] • Apply the this reference [7.1.10] • Add a constructor to initialize a value [7.1.11] • Use access modifiers [7.2.1] • Pass objects to methods [7.2.2] • Return objects from methods [7.2.3] • Use variable argument methods [7.2.4] • Overload constructors [7.2.5] • Overload methods [7.2.6] • Write a class with specified arrays, constructors, and methods [7.2.7] 	<ul style="list-style-type: none"> • Programming with Data Types and Operators [4.3] • Using Strings [4.4] • Using Scanner and Conditional Statements [5.1] • Using Program Control Statements [5.2] • Creating Classes, Objects, and Methods [7.1] • Passing Objects and Overloading Methods [7.2] • Understanding Recursion, the Static Modifier, and Nested Classes [7.3] • Understanding Inheritance [7.4] • Understanding Polymorphism [7.5]

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computing Practice and Programming
	<ul style="list-style-type: none"> • Create static variables [7.3.1] • Use static variables [7.3.2] • Create static methods [7.3.3] • Use static methods [7.3.4] • Create static classes [7.3.5] • Use static classes [7.3.6] • Create linear recursive methods [7.3.7] • Create non-linear recursive methods [7.3.8] • Use the extends keyword to inherit a class [7.4.2] • Use super to call a superclass constructor [7.4.5] • Use super to access superclass members [7.4.6] • Create a multilevel class hierarchy [7.4.7] • Demonstrate understanding of inheritance through the use of applets [7.4.9] • Apply superclass references to subclass objects [7.5.1] • Write code to override methods [7.5.2] • Use dynamic method dispatch to support polymorphism [7.5.3] • Create abstract methods and classes [7.5.4] • Use the final modifier [7.5.6] • Write code for an applet that displays two triangles of different colors [7.5.8] ○ Describe inheritance and how traits are passed from superclasses to subclasses [2.5.4] ○ Describe when to implement procedural abstraction [2.5.5] ○ Understand variables and how they are used in programming [2.10.1] ○ Describe variables [2.13.1] ○ Describe Java simple types [2.13.2] ○ Define arithmetic operators [2.13.3] ○ Describe relational and logical operators [2.13.4] ○ Describe assignment operators [2.13.5] ○ Describe a method, class, and instance [2.14.1] ○ Describe a scenario where an IF control structure would be used [2.14.2] ○ Describe a scenario where a WHILE control structure would be used [2.14.3] ○ Recognize the syntax for a method, class, function, and procedure [2.14.4] 	

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computing Practice and Programming
	<ul style="list-style-type: none"> ○ Describe classes and subclasses [3.1.4] ○ Recognize Java syntax used to correctly create a subclass [3.1.5] ○ Define parameters and how they are used in methods [3.2.1] ○ Understand inheritance [3.2.2] ○ Describe properties of an object [3.2.3] ○ Examine the purpose of a variable [3.2.4] ○ Discuss programming concepts and define terminology [3.2.5] ○ Recognize and describe dot notation [3.5.5] ○ Describe effective placement of methods in a super or subclass [3.6.1] ○ Define the purpose and syntax of a variable [3.8.4] ○ Recognize the syntax to define and test variables [3.8.5] ○ Describe an infinite loop and how to prevent one from occurring [3.10.2] ○ Describe the scope of a local variable in a method [3.10.5] ○ Describe the general form of a Java program [4.2.1] ○ Recognize when an expression requires a type conversion [4.3.5] ○ Describe what happens when a String is modified [4.4.2] ○ Interpret escape sequences in String literals [4.4.4] ○ Recognize the difference between a String and a primitive char data type [4.4.5] ○ Describe why the == operator does not always work when testing String equality [4.4.7] ○ Recognize the correct general form of a class [7.1.1] ○ Describe garbage collection and finalizers [7.1.9] ○ Compare the pros and cons of recursion [7.3.9] ○ Compare and contrast superclasses and subclasses [7.4.3] ○ Describe how inheritance affects member access [7.4.4] ○ Recognize when constructors are called in a class hierarchy [7.4.8] ○ Recognize correct parameter changes in an existing applet [7.4.10] ○ Recognize a correct method override [7.5.5] ○ Explain the purpose and importance of the Object class [7.5.7] ○ Describe object references [7.5.9] 	
Demonstrate good practices in personal information security, using passwords, encryption and secure transactions.	<ul style="list-style-type: none"> ○ Describe the code of ethics and cyber security [1.2.6] 	<ul style="list-style-type: none"> • Introduction [1.2]
Identify interdisciplinary careers that are enhanced by computer science.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9) Strand: Computing Practice and Programming		
Demonstrate dispositions amenable to open-ended problem solving and programming (e.g., comfort with complexity, persistence, brainstorming, adaptability, patience, propensity to tinker, creativity, accepting challenge).		
Collect and analyze data that is output from multiple runs of a computer program.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9) Strand: Computers and Communication Devices		
Recognize that computers are devices that execute programs.		
Identify a variety of electronic devices that contain computational processors.		
Demonstrate an understanding of the relationship between hardware and software.		
Use developmentally appropriate, accurate terminology when communicating about technology.	<ul style="list-style-type: none"> • Describe inheritance and how traits are passed from superclasses to subclasses [2.5.4] • Describe when to implement procedural abstraction [2.5.5] • Describe variables [2.13.1] • Describe Java simple types [2.13.2] • Define arithmetic operators [2.13.3] • Describe relational and logical operators [2.13.4] • Describe assignment operators [2.13.5] • Describe a method, class, and instance [2.14.1] • Describe a scenario where an IF control structure would be used [2.14.2] • Describe a scenario where a WHILE control structure would be used [2.14.3] • Describe input and output [2.14.5] • Define parameters and how they are used in methods [3.2.1] 	<ul style="list-style-type: none"> • Declare Procedures [2.5] • Correlating Java Variables, Data Types, and Expressions with Alice 3 Tools [2.13] • Correlating Java Methods, Classes, and Other Structures with Alice 3 Tools [2.14] • Using Methods, Variables and Parameters [3.2] • Using Loops, Variables, and Strings [3.10] • Use Strings [4.4] • Using Arrays [6.1] • Sorting and Searching [6.2] • Handling Errors [6.3] • Creating Classes, Objects, and Methods [7.1] • Understanding Recursion, the Static Modifier, and Nested Classes [7.3]

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Computers and Communication Devices
	<ul style="list-style-type: none"> • Describe properties of an object [3.2.3] • Describe an infinite loop and how to prevent one from occurring [3.10.2] • Describe what happens when a String is modified [4.4.2] • Describe why the == operator does not always work when testing String equality [4.4.7] • Describe array initialization [6.1.6] • Compare and contrast search and sort algorithms [6.2.5] • Describe the different kinds of errors that can occur and how they are handled in Java [6.3.1] • Describe what exceptions are used for in Java [6.3.2] • Determine what exceptions are thrown for any foundation class [6.3.3] • Describe garbage collection and finalizers [7.1.9] • Compare the pros and cons of recursion [7.3.9] • Demonstrate and explain UML (Unified Modeling Language) class diagrams [7.4.1] • Describe how inheritance affects member access [7.4.4] • Explain the purpose and importance of the Object class [7.5.7] • Describe object references [7.5.9] 	<ul style="list-style-type: none"> • Understanding Inheritance [7.4] • Understanding Polymorphism [7.5]
Apply strategies for identifying and solving routine hardware problems that occur during everyday computer use.		
Describe the major components and functions of computer systems and networks.		
Describe what distinguishes humans from machines, focusing on human intelligence versus machine intelligence and ways we can communicate.		
Describe ways in which computers use models of intelligent behavior (e.g., robot motion, speech and language understanding, and computer vision).		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 2 (Grades 6-9)		Strand: Community, Global, and Ethical Impacts
Exhibit legal and ethical behaviors when using information and technology and discuss the consequences of misuse.	<ul style="list-style-type: none"> ○ Describe the code of ethics and cyber security [1.2.6] 	<ul style="list-style-type: none"> • Introduction [1.2]
Demonstrate knowledge of changes in information technologies over time and the effects those changes have on education, the workplace and society.		
Analyze the positive and negative impacts of computing on human culture.		
Evaluate the accuracy, relevance, appropriateness, comprehensiveness and bias of electronic information sources concerning real-world problems.		
Describe ethical issues that relate to computers and networks (e.g., security, privacy, ownership and information sharing).	<ul style="list-style-type: none"> • Describe the code of ethics and cyber security [1.2.6] <i>[CSTA standard is partially met by the listed objective.]</i> 	<ul style="list-style-type: none"> • Introduction [1.2]
Discuss how the unequal distribution of computing resources in a global economy raises issues of equity, access and power.		



Level 3 (recommended for grades 9–12) Applying concepts and creating real-world solutions

Level 3 is divided into three discrete courses, each of which focuses on different facets of computer science as a discipline. Throughout these courses, students can master more advanced computer science concepts and apply those concepts to develop virtual and real-world artifacts.

Level 3A: (recommended for grades 9 or 10) Computer Science in the Modern World

This course is recommended for all students. Its goal is to solidify students’ understanding of computer science principles and practices so that they can make informed choices and use appropriate computational tools and techniques in whatever career they decide to pursue. They should also appreciate the breadth of computing and its influence in almost every aspect of modern life.

(● = terminal objective: meets all or part of CSTA standard; ○ = enabling objective: an early step that could lead to, but is not sufficient for, meeting CSTA standard)

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computational Thinking
Use predefined functions and parameters, classes and methods to divide a complex problem into simpler parts.	<ul style="list-style-type: none"> ● Simplify programming by creating and calling defined methods [3.6.2] ● Access a minimum of two Java class APIs [4.2.3] ● Create a Driver class [4.2.6] ● Use a method from the Math class [4.3.9] ● Access a Math class method from the Java API [4.3.10] ● Create an object of a class [7.1.2] ● Create a driver class and add instances of Object classes [7.1.6] ● Apply the new operator [7.1.8] ● Pass objects to methods [7.2.2] 	<ul style="list-style-type: none"> ● Defining Methods [3.6] ● Using Object Classes and Driver Classes [4.2] ● Creating Classes, Objects, and Methods [7.1] ● Passing Objects and Overloading Methods [7.2]
Describe a software development process used to solve software problems (e.g., design, coding, testing, verification).		
Explain how sequence, selection, iteration, and recursion are building blocks of algorithms.	<ul style="list-style-type: none"> ● Describe a scenario where an IF control structure would be used [2.14.2] ● Describe a scenario where a WHILE control structure would be used [2.14.3] ● Compare the pros and cons of recursion [7.3.9] 	<ul style="list-style-type: none"> ● Correlating Java Methods, Classes, and Other Structures with Alice 3 Tools [2.14] ● Understanding Recursion, the Static Modifier, and Nested Classes [7.3]
Compare techniques for analyzing massive data collections.		
Describe the relationship between binary and hexadecimal representations.		
Analyze the representation and trade-offs among various forms of digital information.		



CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computational Thinking
Describe how various types of data are stored in a computer system.		
Use modeling and simulation to represent and understand natural phenomena.		
Discuss the value of abstraction to manage problem complexity.	<ul style="list-style-type: none"> Describe when to implement procedural abstraction [2.5.5] 	<ul style="list-style-type: none"> Declare Procedures [2.5]
Describe the concept of parallel processing as a strategy to solve large problems.		
Describe how computation shares features with art and music by translating human intention into an artifact.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Collaboration
Work in a team to design and develop a software artifact.		
Use collaborative tools to communicate with project team members (e.g., discussion threads, wikis, blogs, version control, etc.).		
Describe how computing enhances traditional forms and enables new forms of experience, expression, communication, and collaboration.		
Identify how collaboration influences the design and development of software products.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computing Practice and Programming
Create and organize web pages through the use of a variety of web programming design tools.		
Use mobile devices/ emulators to design, develop, and implement mobile computing applications.		
Use various debugging and testing methods to ensure program correctness (e.g., test cases, unit testing, white box, black box, integration testing)	<ul style="list-style-type: none"> • Test and debug an animation [2.3.8] • Test an animation [2.12.3] • Demonstrate program testing strategies [3.4.1] • Test to ensure application is complete [4.1.4] • Modify a program to execute error free [4.1.6] 	<ul style="list-style-type: none"> • Get Started with Alice 3 [2.1] • Use Procedures and Arguments [2.3] • Develop a Complete Animation [2.12] • Developing and Testing an Application [3.4] • Compiling with Eclipse – A First Program [4.1]
Apply analysis, design, and implementation techniques to solve problems (e.g., use one or more software lifecycle models).	<ul style="list-style-type: none"> • Write a storyboard [2.5.2] • Flowchart a storyboard [2.5.3] • Complete an animation [2.12.2] • Test an animation [2.12.3] • Demonstrate program testing strategies [3.4.1] • Compile an application [4.1.3] • Test to ensure application is complete [4.1.4] • Modify a program to execute error free [4.1.6] <ul style="list-style-type: none"> ○ Recognize phases for developing a software application [3.4.2] ○ List programming tasks to design and implement a game using Java [3.12.4] ○ Identify components of a Java application [4.1.2] 	<ul style="list-style-type: none"> • Declare Procedures [2.5] • Develop a Complete Animation [2.12] • Developing and Testing an Application [3.4] • Creating an Inventory of Java Fundamentals [3.12] • Compiling with Eclipse – A First Program [4.1]
Use Application Program Interfaces (APIs) and libraries to facilitate programming solutions.	<ul style="list-style-type: none"> • Use a method from the Math class [4.3.9] • Access a Math class method from the Java API [4.3.10] • Instantiate (create) a String [4.4.1] • Test Strings with the compareTo() and equals() method [4.4.6] • Use String methods length(), substring(), indexOf(), and charAt() [4.4.8] <ul style="list-style-type: none"> ○ Access a minimum of two Java class APIs [4.2.3] 	<ul style="list-style-type: none"> • Using Object Classes and Driver Classes [4.2] • Programming with Data Types and Operators [4.3] • Using Strings [4.4]
Select appropriate file formats for various types and uses of data.		



CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computing Practice and Programming
Describe a variety of programming languages available to solve problems and develop systems.		
Explain the program execution process.		
Explain the principles of security by examining encryption, cryptography, and authentication techniques.		
Explore a variety of careers to which computing is central.		
Describe techniques for locating and collecting small and large-scale data sets.		
Describe how mathematical and statistical functions, sets, and logic are used in computation.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computers and Communication Devices
Describe the unique features of computers embedded in mobile devices and vehicles (e.g., cell phones, automobiles, airplanes).		
Develop criteria for purchasing or upgrading computer system hardware.		
Describe the principal components of computer organization (e.g., input, output, processing, and storage).		
Compare various forms of input and output.		
Explain the multiple levels of hardware and software that support program execution (e.g., compilers, interpreters, operating systems, networks).		



CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Computers and Communication Devices
Apply strategies for identifying and solving routine hardware and software problems that occur in everyday life.		
Compare and contrast client-server and peer-to-peer network strategies.		
Explain the basic components of computer networks (e.g., servers, file protection, routing, spoolers and queues, shared resources, and fault-tolerance).		
Describe how the Internet facilitates global communication.		
Describe the major applications of artificial intelligence and robotics.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Community, Global, and Ethical Impacts
Compare appropriate and inappropriate social networking behaviors.		
Discuss the impact of computing technology on business and commerce (e.g., automated tracking of goods, automated financial transactions, e-commerce, cloud computing).		
Describe the role that adaptive technology can play in the lives of people with special needs.		
Compare the positive and negative impacts of technology on culture (e.g., social networking, delivery of news and other public media, and intercultural communication).		



CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3A (Grade 9 or 10)		Strand: Community, Global, and Ethical Impacts
Describe strategies for determining the reliability of information found on the Internet.		
Differentiate between information access and information distribution rights.		
Describe how different kinds of software licenses can be used to share and protect intellectual property.		
Discuss the social and economic implications associated with hacking and software piracy.		
Describe different ways in which software is created and shared and their benefits and drawbacks (commercial software, public domain software, open source development).		
Describe security and privacy issues that relate to computer networks.		
Explain the impact of the digital divide on access to critical information.		



Level 3B: (recommended for grades 10 or 11) Computer Science Concepts and Practices

This course is a more in-depth study of computer science and its relation to other disciplines, and contains a significant amount of algorithmic problem solving and related activities. One way to realize this course is by following the Computer Science Principles course (www.apcsprinciples.org). Students should complete this course with a clear understanding of the application of computational thinking to real-world problems.

(● = terminal objective: meets all or part of CSTA standard; ○= enabling objective: an early step that could lead to, but is not sufficient for, meeting CSTA standard)

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Computational Thinking
Classify problems as tractable, intractable, or computationally unsolvable.		
Explain the value of heuristic algorithms to approximate solutions for intractable problems.		
Critically examine classical algorithms and implement an original algorithm.	<ul style="list-style-type: none"> ● Compare and contrast search and sort algorithms [6.2.5] <i>[CSTA standard is partially met by the listed objective.]</i> 	<ul style="list-style-type: none"> ● Sorting and Searching [6.2]
Evaluate algorithms by their efficiency, correctness, and clarity.	<ul style="list-style-type: none"> ● Analyze the Big-O for various sort algorithms [6.2.6] 	<ul style="list-style-type: none"> ● Sorting and Searching [6.2]
Use data analysis to enhance understanding of complex natural and human systems.		
Compare and contrast simple data structures and their uses (e.g., arrays and lists).		
Discuss the interpretation of binary sequences in a variety of forms (e.g., instructions, numbers, text, sound, image).		
Use models and simulations to help formulate, refine, and test scientific hypotheses.		
Analyze data and identify patterns through modeling and simulation.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		
Decompose a problem by defining new functions and classes.	<ul style="list-style-type: none"> • Create an Object class [4.2.5] • Create a Driver class [4.2.6] • Create methods that compile with no errors [7.1.3] • Return a value from a method [7.1.4] • Use parameters in a method [7.1.5] • Create a driver class and add instances of Object classes [7.1.6] • Add a constructor to a class [7.1.7] • Apply the new operator [7.1.8] • Add a constructor to initialize a value [7.1.11] • Use access modifiers [7.2.1] • Pass objects to methods [7.2.2] • Return objects from methods [7.2.3] • Use variable argument methods [7.2.4] • Overload constructors [7.2.5] • Overload methods [7.2.6] • Write a class with specified arrays, constructors, and methods [7.2.7] • Use the extends keyword to inherit a class [7.4.2] • Use super to call a superclass constructor [7.4.5] • Use super to access superclass members [7.4.6] • Create a multilevel class hierarchy [7.4.7] • Apply superclass references to subclass objects [7.5.1] • Write code to override methods [7.5.2] • Use dynamic method dispatch to support polymorphism [7.5.3] • Create abstract methods and classes [7.5.4] • Use the final modifier [7.5.6] ○ Compare and contrast superclasses and subclasses [7.4.3] ○ Describe how inheritance affects member access [7.4.4] 	<p style="text-align: right;">Strand: Computational Thinking</p> <ul style="list-style-type: none"> • Using Object Classes and Driver Classes [4.2] • Creating Classes, Objects, and Methods [7.1] • Passing Objects and Overloading Methods [7.2] • Understanding Inheritance [7.4] • Understanding Polymorphism [7.5]
Demonstrate concurrency by separating processes into threads and dividing data into parallel streams.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Collaboration
Use project collaboration tools, version control systems, and Integrated Development Environments (IDEs) while working on a collaborative software project.		
Demonstrate the software life cycle process by participating on a software project team.		
Evaluate programs written by others for readability and usability.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Computing Practice and Programming
Use advanced tools to create digital artifacts (e.g., web design, animation, video, multimedia).	<ul style="list-style-type: none"> ○ Plan the presentation of a completed animation project [2.12.5] 	<ul style="list-style-type: none"> • Develop a Complete Animation [2.12]
Use tools of abstraction to decompose a large-scale computational problem (e.g., procedural abstraction, object-oriented design, functional design).	<ul style="list-style-type: none"> • Create an Object class [4.2.5] • Create a Driver class [4.2.6] • Create a driver class and add instances of Object classes [7.1.6] • Write a class with specified arrays, constructors, and methods [7.2.7] • Use the extends keyword to inherit a class [7.4.2] • Use super to call a superclass constructor [7.4.5] • Use super to access superclass members [7.4.6] • Create a multilevel class hierarchy [7.4.7] • Apply superclass references to subclass objects [7.5.1] • Write code to override methods [7.5.2] • Use dynamic method dispatch to support polymorphism [7.5.3] • Create abstract methods and classes [7.5.4] ○ Compare and contrast superclasses and subclasses [7.4.3] ○ Describe how inheritance affects member access [7.4.4] 	<ul style="list-style-type: none"> • Using Object Classes and Driver Classes [4.2] • Creating Classes, Objects, and Methods [7.1] • Passing Objects and Overloading Methods [7.2] • Understanding Inheritance [7.4] • Understanding Polymorphism [7.5]
Classify programming languages based on their level and application domain.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Computing Practice and Programming
Explore principles of system design in scaling, efficiency, and security.		
Deploy principles of security by implementing encryption and authentication strategies.		
Anticipate future careers and the technologies that will exist.		
Use data analysis to enhance understanding of complex natural and human systems.		
Deploy various data collection techniques for different types of problems.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Computers and Communication Devices
Discuss the impact of modifications on the functionality of application programs.		
Identify and describe hardware (e.g., physical layers, logic gates, chips, components).		
Identify and select the most appropriate file format based on trade-offs (e.g., accuracy, speed, ease of manipulation).		
Describe the issues that impact network functionality (e.g., latency, bandwidth, firewalls, server capability).		
Explain the notion of intelligent behavior through computer modeling and robotics.		

CSTA Standard	Aligned Oracle Objectives	Oracle Lesson Names
Level 3B (Grade 10 or 11)		Strand: Community, Global, and Ethical Impacts
Demonstrate ethical use of modern communication media and devices.		
Analyze the beneficial and harmful effects of computing innovations.		
Summarize how financial markets, transactions, and predictions have been transformed by automation.		
Summarize how computation has revolutionized the way people build real and virtual organizations and infrastructures.		
Identify laws and regulations that impact the development and use of software.		
Analyze the impact of government regulation on privacy and security.		
Differentiate among open source, freeware, and proprietary software licenses and their applicability to different types of software.		
Relate issues of equity, access, and power to the distribution of computing resources in a global society.		