

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

**Course Description**

AP Computer Science A is a full-year course that meets on a rotating basis for three (3) 55-minute blocks and one (1) 40-minute block for every five (5) day cycle.

Computer science embraces problem-solving, hardware, algorithms, and perspectives that help people utilize computers to solve real-world problems in everyday life. The AP Computer Science A course introduces students to computer science with fundamental topics that include problem-solving, design strategies and methodologies, organization of data, approaches to processing data, analysis of potential solutions, and the ethical and social implications of computing. The course emphasizes both object-oriented and imperative problem-solving and design. These techniques represent proven approaches for developing solutions that can scale up from small, simple problems to large, complex problems. Students will be programming in the language of Java. The course content follows the College Board's Advanced Placement curriculum for the AP test in Computer Science A.

**Course Overview and Pacing Guide**

Unit	Topic	Time Frame
1	Primitive Types	2 Weeks
2	Using Objects	4 Weeks
3	Boolean Expressions and if Statements	3 Weeks
4	Iteration	3 Weeks
5	Writing Classes	4 Weeks
6	Array	3 Weeks
7	ArrayList	3 Weeks
8	2D Array	2 Weeks
9	Inheritance	3 Weeks
10	Recursion	3 Weeks
11	AP Exam Prep	3 Weeks
12	String Manipulations	2 Weeks
13	Design and Object Oriented Principles	3 Weeks

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Computer Science and Design Thinking (Standard 8)</b>	
Core Idea	Performance Expectation
The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.	8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
Successful troubleshooting of complex problems involves multiple approaches including research, analysis, reflection, interaction with peers, and drawing on past experiences.	8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
The scalability and reliability of the Internet are enabled by the hierarchy and redundancy in networks. Network topology is determined by many characteristics.	8.1.12.NI.1: Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	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.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data, and procedures.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users. 8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible. 8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs.
<b>Career Readiness, Life Literacies, and Key Skills (Standard 9)</b>	
Core Idea	Performance Expectation
There are strategies to improve one's professional value and marketability.	9.2.12.CAP.3: Investigate how continuing education contributes to one's career and personal growth.
Career planning requires purposeful planning based on	9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

research, self-knowledge, and informed choices.	those skills.
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas
Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed.	9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice
Laws govern many aspects of computing, such as privacy, data, property, information, and identity. These laws can have beneficial and harmful effects, such as expediting or delaying advancements in computing and protecting or infringing upon people's rights.	9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics 9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users.
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society. These new types of information must be evaluated carefully.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions. 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments. 9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem.
<b>Interdisciplinary Connections</b>	
MA.K-12.1-8: Integrate mathematical skills using variables, structure, and reasoning to create computer programs.	

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Modifications, Accommodations, and Differentiation</b>			
<b>English Language Learners</b>	<b>IEP / 504</b>	<b>At Risk Students</b>	<b>Gifted and Talented</b>
Scaffolding Word walls Sentence/paragraph frames Bilingual dictionaries/translation Think alouds Read alouds Highlight key vocabulary Annotation guides Think-pair- share Visual aides Modeling In-Class-Support Graphic Organizers Note-taking guides Clarify assignments, directions and instructions Extended time Collaboration between ESL and mainstream classroom teachers	Word walls Visual aides Graphic organizers Multimedia Leveled readers Assistive technology Notes/summaries Extended time Answer masking Answer eliminator Highlighter Color contrast In-Class-Support Provide Study Guides Establish Timelines Clarify Assignments, directions and instructions Parent/teacher communication Assist students with steps to complete assignments Provide student with clear expectations and grading criteria Establish procedures for accommodations /modifications for assessments	Teacher tutoring Peer tutoring Study guides Graphic organizers Extended time Parent communication Modified assignments Counseling Verbal reminders Check student independent work Assist student with planning of assignments Provide student with clear expectations and grading criteria Restate and clarify directions and questions	Challenge assignments Enrichment activities Independent research/inquiry Collaborative teamwork Higher level questioning Critical/Analytical thinking tasks Self-directed activities Open forums and debates in the classroom regarding controversial issues

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 1 - Primitive Types	Time Frame - 2 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research and Activities,	

<b>Disciplinary Concept: Primitive Types</b>	
This unit introduces students to the basics of programming in Java, focusing on the use of variables and operators for storing and manipulating primitive data.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures	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
There are strategies to improve one's professional value and marketability	9.2.12.CAP.3: Investigate how continuing education contributes to one's career and personal growth.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Complete a partial line of code</b>	Lesson 1 - Output In Java	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Movie Ratings Project
<b>Determine the results of printing an expression concatenating two strings</b>	Lesson 2 - User Input and Variables	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Movie Ratings Project
<b>Describe different primitive data types that can be used to store numbers in Java</b>	Lesson 3 - Data Types	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Movie Ratings Project
<b>Perform number calculations in Java</b>	Lesson 4 - Number Calculations Lesson 5 - Modular Division	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Movie Ratings Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Understand casting and how to convert one data type to another</b>	Lesson 6 - Numeric Casts	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Movie Ratings Project
---	--------------------------	---

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 2 - Using Objects	Time Frame - 4 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Using Objects</b>	
This unit introduces the use of classes, methods, and objects.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures	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
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Career planning requires purposeful planning based on research, self-knowledge, and informed choices.	9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Understand how data is stored in memory and concatenating string variables with primitives</b>	Lesson 1 - Strings and Class Types Lesson 2 - Escape Char and String Concatenation	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Control Tower Project
<b>Understand methods which can be called on objects of the string data type and the class-object structure in Java</b>	Lesson 3 - String Functions Lesson 4 - Classes and Objects	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Control Tower Project
<b>Use constructors for creating objects of class types intriduced</b>	Lesson 5 - Using Constructors	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Control Tower Project
<b>Write code which creates objects, then call methods to both determine and change the state of the object</b>	Lesson 6 - Using Methods Lesson 7 - Wrapper Classes Lesson 8 - Math Functions	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Control Tower Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 3 - Boolean Expressions and If Statements	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

**Disciplinary Concept: Boolean Expressions and If Statements**

This unit introduces the concepts that programs use to make decisions using the logic of Boolean expressions, If and Else statements.

<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible.
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Use conditional logic in a program with comparison operators</b>	Lesson 1 - Simple Ifs Lesson 2 - Ifs - Making Decisions	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Crack The Code Project
<b>Understand how else statements give greater control over programs</b>	Lesson 3 – Else	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Crack The Code Project
<b>Understand how short-circuit evaluation is used by Java when evaluating Boolean statements</b>	Lesson 4 - Booleans and Truth Tables Lesson 5 - Short Circuit Evaluation	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Crack The Code Project
<b>Learn how to apply De Morgan's law to find equivalent Boolean statements</b>	Lesson 6 - De Morgan's Law	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Crack The Code Project
<b>Compare object data in Java using both the '=='</b>	Lesson 7 - Comparing Objects	Classroom Assignments and Activities, Programming



Carlstadt-East Rutherford Regional School District  
Career and Technology Education Department  
AP Computer Science A Curriculum

operator and the equals method		Labs, Unit 3 Quiz, Unit 3 Test, Crack The Code Project
--------------------------------	--	--

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 4 - Iteration	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Iteration</b>	
This unit introduces the idea of iteration, repeating a process until certain conditions are met. This unit focuses on while and for loops to create algorithms for Strings and numbers. Unit 4 also contains the first College Board lab assignment.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.	8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Use while loops to repeat program code and predict output using a trace table</b>	Lesson 1 - While Loops Lesson 1 1/2 - Tracing Code	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, String Project
<b>Use both selection and iteration to produce a desired output when the main method of a program executes</b>	Lesson 2 - Algorithms for Numbers Lesson 3 - The For Loop	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, String Project
<b>Determine the results of running a code segment with nested iteration statements</b>	Lesson 4 - Algorithms for Strings	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, String Project
<b>Write nested iteration statements to produce a desired output</b>	Lesson 5 - Nested Loops Lesson 6 - Algorithm Efficiency:	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, String Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 5 - Writing Classes	Time Frame - 4 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Writing Classes</b>	
This unit introduces the concepts required for students to write their own classes, which will form the basis for creating more sophisticated programs later in the course. This unit also contains the first of two lessons examining the wider impacts of computer science on society and culture.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. 8.1.12.AP.6: Create artifacts by using procedures within a program, combinations of data, and procedures.
Laws govern many aspects of computing, such as privacy, data, property, information, and identity. These laws can have beneficial and harmful effects, such as expediting or delaying advancements in computing and protecting or infringing upon people's rights.	9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics 9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Write methods that use parameters to send information with a method call</b>	Lesson 1 - Void methods Lesson 2 - Parameters Lesson 3 Parameters	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Fraction Project
<b>Use return statements to allow methods to send information back when they are called</b>	Lesson 4 - Return Methods	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Fraction Project
<b>Create a custom class, which is used to create objects representing vehicles</b>	Lesson 5 - Classes Lesson 6 - Constructors Lesson 7 - Documenting a Class	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Fraction Project
<b>Write a method using iteration and selection which returns values according to a specification</b>	Lesson 8 - Static Vs. Instance Lesson 9 - Wider Impacts of Computing	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Fraction Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 6 - Array	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

**Disciplinary Concept: Array**

This unit introduces the array data structure which can be used to hold multiple primitive values or object references. Students will learn how to write code which traverses an array to search for or manipulate data in the array.

<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society. These new types of information must be evaluated carefully.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions. 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Determine the results of executing a block of code with multiple operators working on array elements</b>	Lesson 1 - One-Dimensional Arrays Lesson 2 - Algorithms – Searching	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Array Statistics Project
<b>Write code which traverses an array and returns information about its contents</b>	Lesson 3 - Arrays of Strings	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Array Statistics Project
<b>Determine the output of a code segment with multiple method calls</b>	Lesson 4 - Algorithms on Arrays	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Array Statistics Project
<b>Write a method which traverses an array and changes the value of its elements by inserting an element</b>	Lesson 5 - The For-Each Loop	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Array Statistics Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 7 - ArrayList	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

**Disciplinary Concept: ArrayList**

This unit introduces the concept of ArrayLists, along with some common search and sort algorithms. This unit also contains the second lesson examining the wider impacts of computer science, with an emphasis on data collection.

<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society. These new types of information must be evaluated carefully.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions. 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Write code which creates and then manipulates the contents of an ArrayList</b>	Lesson 1 - ArrayList Lesson 2 - Traversing ArrayLists	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Game Wheel Project
<b>Determine the output of a code segment which makes multiple calls to ArrayList and String methods</b>	Lesson 3 - Array Algorithms with ArrayLists	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Game Wheel Project
<b>Create a method which implements a selection sort algorithm: manipulating an ArrayList by changing the order of its values</b>	Lesson 4 - Linear Search Lesson 5 - Selection Sort Lesson 6 - Insertion Sort	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Game Wheel Project
<b>Explore the risks to privacy of storing personal data using computer systems</b>	Lesson 7 - Wider Impacts of Data Collection	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Game Wheel Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 8 - 2-D Arrays	Time Frame - 2 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: 2-D Arrays</b>	
This unit introduces the concept of 2-D arrays, along with some algorithms that can be used in conjunction with 2-D arrays.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.8: Evaluate and refine computational artifacts to make them more usable and accessible.
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society. These new types of information must be evaluated carefully.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions. 9.4.12.IML.4: Assess and critique the appropriateness and impact of existing data visualizations for an intended audience.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Complete a partially completed code segment by determining the correct condition to use</b>	Lesson 1 - 2-D Arrays	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Battleship Project
<b>Write a method which is used to create a 2-D array and traverse it to edit the values stored to desired results</b>	Lesson 2 - 2-D Array Algorithms	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Battleship Project
<b>Trace code which traverses a 2-D array and determine what output is when the code is executed</b>	Lesson 2 - 2-D Array Algorithms	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Battleship Project
<b>Write methods which traverse a 2-D array, representing a board in a battleship game.</b>	Battleship Project	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Battleship Project

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 9 - Inheritance	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Inheritance</b>	
This unit introduces the concept of inheritance, the ability to have functionality extend from one class to another. The relationships this process creates are examined through the use of class hierarchy diagrams and examples where objects are declared and initialized as different types.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures	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
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Create a new class</b>	Lesson 1 - Inheritance	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Frisbee Project
<b>Write their code so it interacts with existing code by extending a class for which they are given the source code</b>	Lesson 2 - Inheritance Overriding Methods	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Frisbee Project
<b>Override a method according to specifications by calling a method from the superclass</b>	Lesson 2 - Inheritance Overriding Methods	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Frisbee Project
<b>Make decisions about how to design a class</b>	Lesson 3 - Is-a and Has-a Relationships	Classroom Assignments and Activities, Programming

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

hierarchy for a program	Labs, Unit 9 Quiz, Unit 9 Test, Frisbee Project
-------------------------	---

<b>Unit Name</b>	Unit 10 - Recursion	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Recursion</b>
--

This unit introduces the concept of recursion – defining a process in terms of itself. In practice, this means writing methods that contain calls to themselves to solve a problem.

Core Idea	Performance Expectation (Standard)
Successful troubleshooting of complex problems involves multiple approaches including research, analysis, reflection, interaction with peers, and drawing on past experiences.	8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	8.1.12.AP.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.
Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed.	9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
<b>Determine the output when a recursive method is called</b>	Lesson 1 - Intro to Recursion	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Anagrams Project
<b>Determine how many times a line of code is executed in a certain method call</b>	Lesson 2 - Recursive Functions with Returns	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Anagrams Project
<b>Write a recursive method according to a given specification</b>	Lesson 3 - Binary Search	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Anagrams Project
<b>Test methods with a variety of test cases</b>	Lesson 4 - Merge Sort	Classroom Assignments and Activities, Programming



**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

		Labs, Unit 10 Quiz, Unit 10 Test, Anagrams Project
--	--	--

<b>Unit Name</b>	Unit 11 - AP Exam Prep	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: AP Exam Prep</b>	
The goal of the unit is to allow students to synthesize the material covered throughout the year and review any areas that could use strengthening.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
The scalability and reliability of the Internet are enabled by the hierarchy and redundancy in networks. Network topology is determined by many characteristics.	8.1.12.NI.1: Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	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.4: Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Review Programming Fundamentals, Data Structures, Logic, Algorithms, Object-Oriented Programming, Recursion, and Software Engineering</b>	Diagnostic Exam Released AP Exam Free-Response Questions	Diagnostic Exam Released AP Exam Free-Response Questions AP Classroom Review

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>Unit Name</b>	Unit 12 - String Manipulations	Time Frame - 2 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: String Manipulations</b>	
The goal of the unit is to implement a chatbot and incorporates string manipulations.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures	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
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Career planning requires purposeful planning based on research, self-knowledge, and informed choices.	9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Design a ChatBot to mimic conversation</b>	Activity 1: Try out a ChatBot online to clarify what a ChatBot is and how it is used in computing	Classroom Assignments and Activities, Programming Labs, Magpie Lab
<b>Change the Magpie classes to modifying the Magpie classes</b>	Activity 2: Work with the Magpie code by implementing the existing class, Magpie	Classroom Assignments and Activities, Programming Labs, Magpie Lab

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

	Activity 3: Modify the Magpie classes by working with String methods	
<b>Utilize arrays in the ChatBot for processing</b>	Activity 4: Make the ChatBot more responsive by changing what the user types in Activity 5: Use arrays in the ChatBot for processing	Classroom Assignments and Activities, Programming Labs, Magpie Lab

<b>Unit Name</b>	Unit 13 - Design and Object Oriented Principles	Time Frame - 3 Weeks
<b>Instructional Materials and Resources</b>	Project Stem AP CSA Platform, AP Classroom (collegeboard.org), Introduction to Programming Using Java (on-line textbook), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities	

<b>Disciplinary Concept: Design and Object Oriented Principles</b>	
The goal of the unit is to apply design and object oriented principles to create and play a solitaire game.	
<b>Core Idea</b>	<b>Performance Expectation (Standard)</b>
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs
Digital tools such as artificial intelligence, image enhancement and analysis, and sophisticated computer modeling and simulation create new types of information that may have profound effects on society.	9.4.12.IML.3: Analyze data using tools and models to make valid and reliable claims, or to determine optimal design solutions.
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas

<b>Student Learning Objectives (Knowledge and Skills)</b>	<b>Suggested Tasks/Activities</b>	<b>Evidence of Learning (Assessments)</b>
<b>Complete a Card class that will be used to create card objects</b>	Activity One: Build the Card class Activity Two: Build a Deck class to hold Cards Activity Three: Shuffle the cards in the deck	Classroom Assignments and Activities, Programming Labs, Elevens Lab
<b>Design and implement the Card and Deck</b>	Activity Four: Add a Shuffle method to the Deck class.	Classroom Assignments and Activities, Programming

**Carlstadt-East Rutherford Regional School District**  
**Career and Technology Education Department**  
**AP Computer Science A Curriculum**

<b>classes</b>	Activity Five: Testing with Assertions Activity Six: How to play Elevens. Activity Seven: Designing the ElevensBoard class	Labs, Elevens Lab
<b>Explore how inheritance can be used to reuse the code that is common to all games without rewriting it</b>	Activity Eight: Create different games. Activity Nine: Complete the ElevensBoard class. Activity Ten: ThirteensBoard Activity Eleven: Simulation of Elevens	Classroom Assignments and Activities, Programming Labs, Elevens Lab