FAIRFIELD
PUBLIC SCHOOLS

Advanced Placement Computer Science A

Course Information

Grade(s):	10,11,12
Discipline/Course:	Business
Course Title:	AP Computer Science A (Java)
Prerequisite(s):	APCS Principles (B or better) or Programming and Game Design (B or better)
Course Description: *Program of Studies*	The Advanced Placement Computer Science Course offers advanced students an opportunity to complete college-level work in high school. Utilizing the Java programming language, the course provides an introduction to the fundamental concepts of object-based analysis (OOA), design (OOD), and programming (OOP), and how object-oriented languages differ from procedural languages. Students will work on a wide variety of interesting and challenging problems that will be used as a context to focus on problem-solving skills and higher-level thinking. The topics covered include the concepts of abstraction, encapsulation, modularity, inheritance, analysis of algorithms, and polymorphism. The course will focus on CS-1 material (A curriculum). Students are expected to take the AP test in May.
Course Essential Questions:	● What is the value of computers in today's society? ● What are the strengths and limitations of computers? ● How does software affect our lives? ● How do we break down a problem?
Course Enduring Understandings:	The Advanced Placement Computer Science Course offers advanced students an opportunity to complete college-level work in high school. Utilizing the Java programming language, the course provides an introduction to the fundamental concepts of object-based analysis (OOA), design (OOD), and programming (OOP), and how object-oriented languages differ from procedural languages. Students will work on a wide variety of interesting and challenging problems that will be used as a context to focus on problem-solving skills and higher-level thinking. The topics covered include the concepts of abstraction,

	encapsulation, modularity, inheritance, analysis of algorithms, and polymorphism. The course will focus on CS-1 material (A curriculum). Students are expected to take the AP test in May.
Duration and Credit:	1 year / 1.0 credits
Course Materials/Resources:	Access to College Board account Access to PC/Laptops that support Java SDK
FPS Course Academic Expectation(s):	Creating and Constructing The student transfers or extends constructed knowledge to draft and develop ideas, claims, products, or solutions. Synthesizing and Evaluating The student analyzes and interprets text, phenomena, or strategies to critically evaluate and synthesize information.
Year at a Glance (Units):	Unit 1: Introduction to Computer Careers (2 weeks) Unit 2: Primitive Types (3 weeks) Unit 3: Using Objects (3 weeks) Unit 4: Boolean Expressions and if Statements (4 week) Unit 5: Iteration (8 weeks) Unit 6: Writing Classes (8 week) Unit 7: Array (1.5 weeks) Unit 8: ArrayList (3 weeks) Unit 9: 2D Array (3 weeks) Unit 10: Inheritance (5 weeks) Unit 11: Recursion (1.5 weeks)

Units

Unit Number and Title:	Unit 1: Introduction to Computer Careers
Duration:	2 weeks
Resource(s):	Online Career Databases Technology Resources Software: word processing, spreadsheet, presentation
Overview	To provide students with educational and career path options during and after high school.
Learning Goals	
Standard(s):	State of Connecticut Curriculum Frameworks Connecticut State Standards are met in the following areas: EKS.03.01 - Demonstrate use of relational expressions such as: equal to, not equal, greater than, less than, etc. EKS.05.02 - Analyze elements of a problem to develop creative solutions. EKS.05.04 Create ideas, proposals, and solutions to problems 21st Century Skills/International Society for Technology in Education National Business Education Association (NBEA) Standards
Essential Question(s):	● What challenges do computer professionals face in today's world? ● What will the computer industry look like in 10 years?
Enduring Understanding(s):	● Define different careers associated with computers. ● Define the requirements and education necessary for these careers.
Learning Goal(s): *Students will be able to use*	Students will be able to: ● use analytical skills and support conclusions with specificity.

their learning to:	● access and research information using the Internet. ● display creative thinking, problem-solving, and decision making. ● organize and maintain files. ● use computers to process information.

Unit Number and Title:	Unit 2 - Primitive Types
Duration:	3 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	In this unit, students develop an understanding of data encapsulation and object-oriented design decisions.
Learning Goals	
Standard(s):	Standards ● BIG IDEA 1 ● BIG IDEA 2 ● BIG IDEA 3
Essential Question(s):	● How can we use programs to solve problems? ● In what ways are numbers used in the programs and apps you use most often? ● How are mathematical concepts being used in the programs and apps that you use most often?
Enduring Understanding(s):	● Some objects or concepts are so frequently represented that programmers can draw upon existing code that has already been tested, enabling them to write solutions more quickly and with a greater degree of confidence. ● The way variables and operators are sequenced and combined in an expression determines the computed result.
Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● call system class methods to generate output to the console. ● create string literals. ● evaluate arithmetic expressions in a program code. ● evaluate what is stored in a variable as a result of an expression with an assignment statement. ● evaluate arithmetic expressions that use casting

Unit Number and Title:	Unit 3 –Using Objects
Duration:	4 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	In the second unit, students used primitive types to represent real-world data and determined how to use them in arithmetic expressions to solve problems. This unit introduces a new type of data: reference data. Reference data allows real-world objects to be represented in varying degrees specific to a programmer's purpose. This unit builds on students' ability to write expressions by introducing them to Math class methods to write expressions for generating random numbers and other more complex operations. In addition, strings and the existing methods within the String class are an important topic within this unit.
Learning Goals	
Standard(s):	● BIG IDEA 1 Modularity MOD ● BIG IDEA 2 Variables VAR ● BIG IDEA 3 Control CON
Essential Question(s):	● How can we simulate election results using existing program code? ● How are appropriate variables chosen to represent a remote control? ● How do the games we play simulate randomness?
Enduring Understanding(s):	● Some objects or concepts are so frequently represented that programmers can draw upon existing code that has already been tested, enabling them to write solutions more quickly and with a greater degree of confidence. ● Some objects or concepts are so frequently represented that programmers can draw upon existing code that has already been tested, enabling them to write solutions more quickly and with a greater degree of confidence

Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● explain the relationship between a class and an object. ● identify, using its signature, the correct constructor being called. ● define variables of the correct types to represent reference data. ● call non-static void methods without parameters. ● call non-static void methods with parameters. ● for string class: a. Create String objects. b. Call String methods ● for wrapper classes: a. Create Integer objects. b. Call Integer methods. c. Create Double objects. D. Call Double methods. ● call static methods. ● evaluate expressions that use the math class methods.

Unit Number and Title:	Unit 4 –Boolean Expressions and if Statements
Duration:	4 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	In this unit, students create one-dimensional arrays to store lists of primitive values and object references.
Learning Goals	
Standard(s):	● BIG IDEA 1 Control CON
Essential Question(s):	● How can you use different conditional statements to write a pick-your-own-path interactive story? ● Why is selection a necessary part of programming languages?
Enduring Understanding(s):	● The way variables and operators are sequenced and combined in an expression determines the computed result. ● 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 Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● evaluate Boolean expressions that use relational operators in program code. ● represent branching logical processes by using conditional statements. ● represent branching logical processes by using nested conditional statements. ● compare and contrast equivalent Boolean expressions. ● compare object references using Boolean expressions in program code.

Unit Number and Title:	Unit 5 – Iteration
Duration:	5 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	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.
Learning Goals	
Standard(s):	● BIG IDEA 1 Control CON
Essential Question(s):	● 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?
Enduring Understanding(s):	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 Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● represent iterative processes using a while loop ● 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 ● represent iterative processes using a for loop. ● for algorithms in the context of a particular specification that involves String objects: a.)

	Identify standard algorithms. b.) Modify standard algorithms. c.) Develop an algorithm. ● represent nested iterative processes. ● compute statement execution counts and informal run-time comparison of iterative statements.

Unit Number and Title:	Unit 6 – Writing Classes
Duration:	8 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	This unit will pull together information from all previous units to create new, user-defined reference data types in the form of classes. The ability to accurately model real-world entities in a computer program is a large part of what makes computer science so powerful. This unit focuses on identifying appropriate behaviors and attributes of real-world entities and organizing these into classes. Students will build on what they learn in this unit to represent relationships between classes through hierarchies, which appear in Unit 9. The creation of computer programs can have extensive impacts on societies, economies, and cultures. The legal and ethical concerns that come with programs and the responsibilities of programmers are also addressed in this unit.
Learning Goals	
Standard(s):	● BIG IDEA 1 Modularity MOD ● BIG IDEA 2 Variables VAR ● BIG IDEA 3 Impact of Computing IOC
Essential Question(s):	● How can using a model of traffic patterns improve travel time? ● How can programs be written to protect your bank account balance from being inadvertently changed? ● What responsibility do programmers have for the consequences of programs they create, whether intentional or not?
Enduring Understanding(s):	● Programmers use code to represent a physical object or nonphysical concept, real or imagined, by defining a class based on the attributes and/or behaviors of the object or concept. ● To find specific solutions to generalizable problems, programmers include variables in their code so that the same algorithm runs using different input values.

	● While programs are typically designed to achieve a specific purpose, they may have unintended consequences.
Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● designate private visibility of instance variables to encapsulate the attributes of an object. ● define instance variables for the attributes to be initialized through the constructors of a class. ● describe the functionality and use of program code through comments ● define behaviors of an object through non-void methods without parameters written in a class. ● define behaviors of an object through void methods with or without parameters written in a class. ● define behaviors of an object through non-void methods with parameters written in a class. ● define behaviors of a class through static methods. ● define the static variables that belong to the class. ● explain where variables can be used in the program code. ● evaluate object reference expressions that use the keyword this. ● explain the ethical and social implications of computing systems.

Unit Number and Title:	Unit 7 – Array
Duration:	1.5 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	This unit focuses on data structures, which are used to represent collections of related data using a single variable rather than multiple variables. Using a data structure along with iterative statements with appropriate bounds will allow for similar treatment to be applied more easily to all values in the collection. Just as there are useful standard algorithms when dealing with primitive data, there are standard algorithms to use with data structures. In this unit, we apply standard algorithms to arrays; however, these same algorithms are used with ArrayLists and 2D arrays as well. Additional standard algorithms, such as standard searching and sorting algorithms, will be covered in the next unit.
Learning Goals	
Standard(s):	BIG IDEA 1 Variables VAR BIG IDEA 2 Control CON
Essential Question(s):	● How can programs leverage volcano data to make predictions about the date of the next eruption? ● How can knowing standard algorithms be useful when solving new problems?
Enduring Understanding(s):	● To manage large amounts of data or com ● plex relationships in data, programmers write code that groups the data together into a single data structure without creating individual variables for each value. ● 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 Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● represent collections of related primitive or object reference data using one dimensional (1D) array objects

	● traverse the elements in a 1D array. ● traverse the elements in a 1D array object using an enhanced for loop ● for algorithms in the context of a particular specification that requires the use of array traversals: a.) Identify standard algorithms. b.) Modify standard algorithms. c.) Develop an algorithm

Unit Number and Title:	Unit 8 - ArrayList
Duration:	3 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	As students learned in Unit 6, data structures are helpful when storing multiple related data values. Arrays have a static size, which causes limitations related to the number of elements stored, and it can be challenging to reorder elements stored in arrays. The ArrayList object has a dynamic size, and the class contains methods for insertion and deletion of elements, making reordering and shifting items easier. Deciding which data structure to select becomes increasingly important as the size of the data set grows, such as when using a large real-world data set. In this unit, students will also learn about privacy concerns related to storing large amounts of personal data and about what can happen if such information is compromised.
Learning Goals	
Standard(s):	BIG IDEA 1 Variables VAR BIG IDEA 2 Control CON BIG IDEA 3 Impact of Computing IOC
Essential Question(s):	● Why is an ArrayList more appropriate for storing your music playlist, while an array might be more appropriate for storing your class schedule? ● How can we use statement execution counts to choose appropriate algorithms? ● What personal data is currently being collected, and how?
Enduring Understanding(s):	To manage large amounts of data or complex relationships in data, programmers write code that groups the data together into a single data structure without creating individual variables for each value.
Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● represent collections of related object reference data using ArrayList objects. ● for ArrayList objects: a. Traverse using a for or while loop b. Traverse using an enhanced for

	loop ● for algorithms in the context of a particular specification that requires the use of ArrayList traversals: § Identify standard algorithms. § Modify standard algorithms. § Develop an algorithm. ● apply sequential/linear search algorithms to search for specific information in array or ArrayList objects. ● compute statement execution counts and informal run-time comparison of sorting algorithms. ● apply selection sort and insertion sort algorithms to sort the elements of array or ArrayList objects. ● explain the risks to privacy from collecting and storing personal data on computer systems.

Unit Number and Title:	Unit 9 - 2D Arrays
Duration:	3 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	In Unit 6, students learned how 1D arrays store large amounts of related data. These same concepts will be implemented with two-dimensional (2D) arrays in this unit. A 2D array is most suitable to represent a table. Each table element is accessed using the variable name and row and column indices. Unlike 1D arrays, 2D arrays require nested iterative statements to traverse and access all elements. The easiest way to accomplished this is in row-major order, but it is important to cover additional traversal patterns, such as back and forth or column-major.
Learning Goals	
Standard(s):	BIG IDEA 1 Variables VAR BIG IDEA 2 Control CON
Essential Question(s):	● Why might you want to use a 2D array to store the spaces on a game board or the pixels in a picture, rather than a 1D array or ArrayList? ● Why does the order in which elements are accessed in 2D array traversal matter in some situations?
Enduring Understanding(s):	Represent collections of related primitive or object reference data using two-dimensional (2D) array objects.
Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● represent collections of related object reference data using ArrayList objects. ● for 2D array objects: a.) Traverse using nested for loops; b.) Traverse using nested enhanced for loops. ● for algorithms in the context of a particular specification that requires the use of 2D Array traversals: § Identify standard algorithms. § Modify standard algorithms. § Develop an algorithm.

Unit Number and Title:	Unit 10 - Inheritance
Duration:	5 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	Creating objects, calling methods on the objects created, and being able to define a new data type by creating a class are essential understandings before moving into this unit. One of the strongest advantages of Java is the ability to categorize classes into hierarchies through inheritance. Certain existing classes can be extended to include new behaviors and attributes without altering existing code. These newly created classes are called subclasses. In this unit, students will learn how to recognize common attributes and behaviors that can be used in a superclass and will then create a hierarchy by writing subclasses to extend a superclass. Recognizing and utilizing existing hierarchies will help students create more readable and maintainable programs
Standard(s):	BIG IDEA 1 Modularity MOD
Essential Question(s):	● How might the use of inheritance help in writing a program that simulates crops being grown in a virtual world? ● How does inheritance make programs more versatile?
Enduring Understanding(s):	When multiple classes contain common attributes and behaviors, programmers create a new class containing the shared attributes and behaviors forming a hierarchy. Modifications made at the highest level of the hierarchy apply to the subclasses.
Learning Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● create an inheritance relationship from a subclass to the superclass. ● define reference variables of a superclass to be assigned to an object of a subclass in the same hierarchy.

	● call methods in an inheritance relationship. ● call object class methods through inheritance.

Unit Number and Title:	Unit 11 - Recursion

Duration:	1.5 weeks
Resource(s):	Access to College Board account Access to PC/Laptops that support Java SDK
Unit Overview:	Sometimes a problem can be solved by solving smaller or simpler versions of the same problem rather than attempting an iterative solution. This is called recursion, and it is a powerful math and computer science idea. In this unit, students will revisit how control is passed when methods are called, which is necessary knowledge when working with recursion. Tracing skills introduced in Unit 2 are helpful for determining the purpose or output of a recursive method. In this unit, students will learn how to write simple recursive methods and determine the purpose or output of a recursive method by tracing.
Standard(s):	BIG IDEA 1 Control CON
Essential Question(s):	● What real-world processes do you follow that are recursive in nature? ● Why do programmers sometimes prefer using recursive solutions when sorting data in a large data set?
Enduring Understanding(s):	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 Goal(s): *Students will be able to use their learning to:*	Students will be able to: ● Determine the result of executing recursive methods. ● Apply recursive search algorithms to information in String, 1D array, or ArrayList objects.

Reff: AP® Computer Science A COURSE AND EXAM DESCRIPTION, Effective Fall 2019

https://apstudents.collegeboard.org/sites/default/files/2019-05/ap-computer-science-a-course-and-exam-description.p