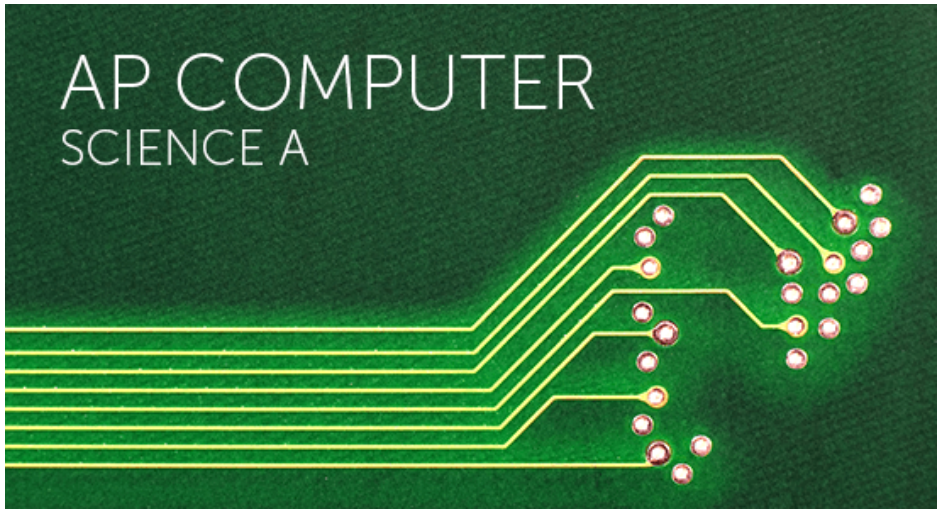
AP COMPUTER SCIENCE A CURRICULUM



Grade Level(s): 10-12

Curriculum Author(s): Raymond Robillard

Course Description: 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 (data structures), approaches to processing data (algorithms), analysis of potential solutions, and the ethical and social implications of computing. Students will focus on both object-oriented and imperative problem solving and design. The AP Computer Science A course curriculum is compatible with many CS1 courses in colleges and universities. Students will be able to obtain college credits by successfully meeting a benchmark score on the Advanced Placement Exam. The programming language used in the course is JAVA.

Year At A Glance			
Unit Title	**Overarching Essential Question**	**Overarching Enduring Understanding**	**Vision of A Learner "I Can" Statements**
Data	• How do we express values and text in Java? • How does a program interact with a user through input and output streams? • What types of data can Java work with?	• All data, at the most basic level, can be represented as a series of 0's and 1's. • Real-world data, whether quantitative or qualitative, can be expressed through the use of 8 primitive data types which are the building blocks for object-based data.	TCC2(9-12)
Branches and Loops	• Why is it beneficial to automate a repeated process? • How do we repeat a command in Java as many times as we like? • How can we maintain a procedure in Java until a condition is met?	• Computers model decision making through the use of conditional statements. • To repeat an action some number of times, a loop can be used in place of many consecutive, repeated, commands. • Most real world phenomena can be modeled with algorithms involving a combination of sequencing, selection, and iteration.	TI1(9-12)
More Java Features	• How can we simplify some of the more common commands used in Java programming? • What are other ways to set up conditionals and loops? • How can we program for non-binary situations?	• When a process is common enough, shorthand notation and alternate techniques can streamline the communicating and implementing of this process. In this unit, we learn alternate, and more efficient, ways to handle some common Java protocols.	TCC3(9-12), TI2(9-12)

Object Oriented Programming	• How does Java differ from older programming languages like Basic? • What is Object Oriented Programming and what are its advantages? • How do we describe objects and classes of objects?	• Real world objects can be represented abstractly in a computer program by defining their traits(instance variables), functions(methods), and a basis for their creation(constructors). • Multiple instances of the same object can behave differently and have different traits. • Object Oriented Programming allows for greater/easier control of data structures.	TCC4(9-12)
Arrays	• How can we create visual representations in a Java program? • Why is using an array better than using a large number of variables? • What types of data can be stored in an array?	• An Array is a data collection tool that allows for organization of real world lists such as rosters, statistics, testing results, etc. • Variables can be referenced through the name of a collection object and the index of that object more easily than through the use of numerous variable names. • A multi-dimensional array is structured as an array of an array.	TCC3(9-12), AA4(9-12)
Advanced Object Oriented Programming	• How does abstraction improve the functionality, and readability, or a program? • What types of real-world scenarios can be represented through the use of a hierarchy (and by extension, parent and child classes)? • How are abstract classes and interfaces similar and how are they different?	• Inheritance allows for related representations of objects to be created without repetition of common traits or procedures. • Abstraction and Polymorphism allow us to have, analyze, and manipulate generalized collections of data	TCC4(9-12), AA2(9-12)
Recursion	• What are the key components of any recursive process? • What are some advantages and	• Recursion involves a base case, which describes the situation where recursion concludes, a	AA1(9-12)

	disadvantages to using recursion?	recursive rule, and typically, some return statement to connect instances of the recursive process. ● Recursion can solve the same problems that a loop can, however, usually in a more efficient way.	
Searching and Sorting	● What makes a searching or sorting algorithm efficient? ● How do the gains in efficiency of a sorting algorithm balance against the added complexity to program it? ● Can all searching and sorting problems be solved with an algorithm?	● There are dozens of various searching and sorting algorithms, each with distinct advantages and disadvantages. ● Almost every occupation requires the ability to search or sort data in a meaningful way.	TCC3(9-12), TCC4(9-12)

Unit 1 - Data

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-DA-07 Represent data using multiple encoding schemes. (P4.0)
2-AP-11 Create clearly named variables that represent different data types and perform operations on their values. (P5.1, P5.2)
3A-DA-09 Translate between different bit representations of real-world phenomena, such as characters, numbers, and images. (P4.1)
3A-DA-10 Evaluate the tradeoffs in how data elements are organized and where data is stored. (P3.3)

Vision of A Learner Attributes:
TCC2(9-12): I can evaluate evidence from multiple perspectives, and recognize their limitations and implications, in order to justify new conclusions.

Understandings: Students will understand that...
- All data, at the most basic level, can be represented as a series of 0's and 1's.
- Real-world data, whether quantitative or qualitative, can be expressed through the use of 8 primitive data types which are the building blocks for object-based data.

Essential Questions:
- How do we express values and text in Java?
- How does a program interact with a user through input and output streams?
- What types of data can Java work with?

Students will know...
- There are several primitive data types with different purposes, using varying degrees of memory.
- All data is expressed as either primitive data types or objects.
- Division performed on two ints produces an answer with no remainder.
- The reserved word "final" will tell the compiler not to change the value of a variable.
- A class is a description of a kind of object. It describes what an object of a particular type is made of(data and methods).
- Creating an object reference is not the same as creating an object.
- NaN stands for "Not a Number".

Students will be able to...
- Make appropriate declarations of variables, avoiding reserved words.
- Perform operations on numerical variables, including writing simple counting programs.
- Write simple programs that use the scanner class to input data.
- Interpret a number in scientific notation as a really large or really small number.
- Utilize the sqrt() method to take the square root of floating point values.
- Utilize methods of the Math class to solve problems involving trigonometry.

Key Vocabulary: Primitive Data Type, Bit, Byte, Short, Int, Long, Float, Double, Char, Boolean, Object, Literal, Scientific Notation, Variable, Declaration, Nested ( ), Subexpressions, Constant, Calling a method, Dot notation, Package, Wrapper class, Input stream, Output stream, Sqrt(), NaN, Type casting

Assessment Evidence

Performance Tasks:	**Other Evidence:**
Unit 1 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts. The options include programs to: ● Solve problems utilizing the Law of Cosines ● Calculate an original cost, sales tax, and total cost for varying amounts of purchased coffee ● Estimate a population of a city, given a certain rate and amount of time, using an exponential model ● Determine interest rates on loans	Unit 1 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

HELLO WORLD:
Students will write their first program, one that displays output on the screen (specifically, “Hello World”)
MULTIPLICATION FORMULA PROGRAM:
Students will write a program that should take 2 input values from the user and output the product of those numbers. Note: Be sure your program clearly communicates with the user what input it needs and what it does with those values.
MULTIPLICATION FORMULA PROGRAM:
Students will write a program that should take 2 input values from the user and output the product of those numbers. Note: Be sure your program clearly communicates with the user what input it needs and what it does with those values.
MONEY CONVERTER:
Students will write a program using the remainder operator to take in an input of some number of pennies and report out how many dollars,

quarters, dimes, nickels, and pennies this would be most efficiently converted to.
PYTHAGOREAN THEOREM CALCULATOR:
Students will write code that, given two user-entered values, will find the length of the hypotenuse of a right triangle and output that value to the user. Note: The output can be written as “the square root of ___” to avoid the use of decimal digits.
SLOPE FORMULA CALCULATOR:
Students will write code that, given four user-entered values, will find the slope of the line that contains two ordered pairs, or displays undefined, if appropriate.

Teacher Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 2 - Branches and Loops

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-AP-12 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals. (P5.1, P5.2)
3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. (P5.2)

Vision of A Learner Attributes:
TI1(9-12): I can implement a realistic plan and adapt when necessary to achieve my goals.

Understandings: Students will understand that...
- Computers model decision making through the use of conditional statements.
- To repeat an action some number of times, a loop can be used in place of many consecutive, repeated, commands.
- Most real world phenomena can be modeled with algorithms involving a combination of sequencing, selection, and iteration.

Essential Questions:
- Why is it beneficial to automate a repeated process?
- How do we repeat a command in Java as many times as we like?
- How can we maintain a procedure in Java until a condition is met?

Students will know...
- A boolean expression can take on one of two values: true or false.
- Short circuit evaluation means that sometimes Java only needs to check the first relational expression to evaluate the value of a more complex boolean expression.
- The "!" operator has precedence over other operators.
- Random numbers generated by the Random() method are only pseudorandom generated by a seed value and an algorithm.

Students will be able to...
- Write if and if-then statements as part of a Java program.
- Construct a flowchart to model the workings of a Java program with multiple branches and loops.
- Use the "&&" and "||" operators to write complex boolean expressions.
- Use the "!" operator to reverse the value of boolean expressions.
- Use "while" loops, with and without iteration in Java programs.
- Use the Random() method of the Random Class to generate random digits for use in Java programs.

Key Vocabulary: Two-way Decision(Binary Decision), If Statement, Block Statement, Boolean Expression, Flowchart, Nested, Relational Expression, Relational Operator, Logical Operator, "&&" and "||" operators, Truth Table, "!" operator, Loop, While Statement, Loop control variable, Counting loop, Iteration, Infinite Loop, Nested Loops, Sentinel Controlled Loops, Result Controlled Loops, Random(), Random Number Generator,

Uniformly Distributed, Pseudorandom, Seed

Assessment Evidence

Performance Tasks:	**Other Evidence:**
Unit 2 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts. The options include programs to: ● Determine if a given year is/was a leap year Calculate total sales and commission for an input of sales amounts ● Utilize a flow chart to determine why a lamp isn't working ● Calculate the standard deviation and variance from a set of data	Unit 2 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

IMPROVED SLOPE FINDER PROGRAM: Students will revise the previous written Slope Finder program to now handle the corner case of an undefined slope for a vertical line.
GUESS WHO ACTIVITY:
Students will write a program that, given the least number of pieces of information, stored as boolean variables, will uniquely identify each member of the class.
BINARY MAGIC TRICK PROGRAM: Students will write a program to carry out a "guess your number" trick that utilizes base 2 representations of numbers.
IMPROVED PYTHAGOREAN THEOREM FINDER PROGRAM:
Students will revise the previous written Pythagorean Theorem program to now consider non-right triangles to indicate what type of triangle exists based on user input of side lengths
CHECK DIGIT PROGRAM:
Students will write code that takes an input of an ISBN Number(Semester Course) or a Credit Card Number(AP) and uses the check digit scheme to determine if the inputted number is legitimate. If it is not, the program should state what the correct check digit should be.
RANDOM CARD MAKER:
Students will write a program that generates a random int from 1-13 (1= ace, 11=jack, 12=queen, 13=king) and a second random int from 1-4 (for the 4 suits). The program will tell the user what card was randomly drawn using if statements to convert numbers to ranks and suits.

Teacher Resources: http://chortle.ccsu.edu/Java5/index.html#07 Java Runtime Environment Java Development Kit

Unit 3 - More Java Features

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-AP-10 Use flowcharts and/or pseudocode to address complex problems as algorithms. (P4.4, P4.1)
3A-AP-15 Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made. (P5.2)

Vision of A Learner Attributes:
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
TI2(9-12): I can evaluate my objectives and a variety of credible resources to find the best solutions for any challenge.

Understandings: Students will understand that...
- When a process is common enough, shorthand notation and alternate techniques can streamline the communicating and implementing of this process. In this unit, we learn alternate, and more efficient, ways to handle some common Java protocols.

Essential Questions:
- How can we simplify some of the more common commands used in Java programming?
- What are other ways to set up conditionals and loops?
- How can we program for non-binary situations?

Students will know...
- Postfix operators increment after use, where prefix operators increment before use.
- Truth tables can provide valuable clarity into the possible outcomes from a boolean expression.
- Short circuit evaluation means that a program often only has to evaluate part of a boolean expression.
- There is a greater potential for programming bugs when using bottom-driven loops.

Students will be able to...
- Utilize increment operators to simplify programs.
- Write For statements as part of a Java program.
- Evaluate the truth of a boolean expression using a truth table.
- Write Switch statements as part of a Java program.
- Write Do and Do-while statements as part of a Java program.

Key Vocabulary: Postfix operator, Prefix operator, Short-circuit evaluation, Pure function, Associativity, Truth table, DeMorgan's Rules, For loop, Top-driven loop, Scope, Scope rule, Switch, Conditional operator, Conditional expression, Break, Do statement, Do-while statement, Bottom-driven loops

Assessment Evidence	
Performance Tasks: Unit 3 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts. The options include programs to: ● Calculates the number of primes between 2 and N and displays them on the screen ● Converts a Roman Numeral into a standard Arabic Numeral ● Determine the authenticity of a triangle using the Triangle Inequality Theorem and input side lengths ● Simulate a playing of the game “Buzz”	**Other Evidence:** Unit 3 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

SOUNDEX CODING SYSTEM:
Google uses a system known as the Soundex Coding System to accurately determine the intention of a search request despite spelling errors. Students will write a program that asks the user for a word, then uses the soundex coding system to code it. The program should return the computed soundex code. For example, for Kobey Briant it should return K116.
IMPROVED LOOP PROGRAM:
Students will revisit any previously written program that required a loop and modify it to include a for loop instead.
CAESAR CYPHER:
Students will write a Caesar Cypher program. This program would take in a user input of text as well as a number(x) for the displacement of the letters in the alphabet. The program will use a switch statement to code the text by swapping out letters for the ones x spaces over in the alphabet. For example, if x was 3, a->d, b->e, c->f, and so on. The program should then report what the coded message would be.
INPUT FILE PROMPT:
Students will write a class with a main() method that asks the user for the name of the input file to be read by the program. The file name must be in the correct format. File names consist of two parts, a name and an extension. The name must not contain spaces. The extension must be ".dat" (for "data"). The user is repeatedly asked for a file name until it meets the requirement.

Teacher Resources:

http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 4 - Object Oriented Programming

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-AP-14 Create procedures with parameters to organize code and make it easier to reuse. (P4.1, P4. 3)
3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. (P3.2)
3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. (P5.2)

Vision of A Learner Attributes:
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.

Understandings: Students will understand that...
- Real world objects can be represented abstractly in a computer program by defining their traits(instance variables), functions(methods), and a basis for their creation(constructors).
- Multiple instances of the same object can behave differently and have different traits.
- Object Oriented Programming allows for greater/easier control of data structures.

Essential Questions:
- How does Java differ from older programming languages like Basic?
- What is Object Oriented Programming and what are its advantages?
- How do we describe objects and classes of objects?

Students will know...
- An object has identity, state, and behavior.
- A class is a description of a possible object whereas an object is a unique instance of a class.
- A program can execute a static method without first creating an object.
- An object reference is assigned to a variable, not the object itself.
- A reference variable that contains no object is “null”.
- String objects are immutable.
- Private variables and methods may only be used within their assigned class whereas public variables and methods are accessible to all classes in a package.

Students will be able to...
- Create point objects for use in Java programs.
- Created new strings through various methods of String objects.
- Write class and method definitions to perform various tasks.
- Create classes that contain instance variables, constructors, and methods.
- Model a car’s gas mileage with a Java program involving multiple classes, objects, and methods.
- Model a series of bank accounts with a Java program involving multiple classes, objects, and methods.
- Determine whether a variable or method should be declared public or private.

• Objects may contain other objects.	

Key Vocabulary: Object, Class, Identity, State, Behavior, Instantiation, Members of an object, Dot Notation, Static, Constructor, Immutable, Object Reference Variable, Garbage Collector, String Literal, Point class, toString(), equals(), Alias, Method call, Type cast, The null value, concat(), Overloaded, Parameter, Write-once objects, trim(), charat(), substring(), startswith(), Cascade, String Indexing, Return, Void, Default constructor, Private, Public, Encapsulation, Passing a value, Scope, Signature, Call by value

Assessment Evidence

Performance Tasks:
Unit 4 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts.
The options include programs to:

- Convert weights from Earth weight to equivalent weight on other planets
- Simulates random rolls of any number and type of dice
- Replicates college meal cards and allows for the adding and subtracting of credit
- Verify whether a password meets some number of rules

Other Evidence:
Unit 4 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

NEW CLASS OF OBJECTS PROJECT:
Students will create a class of some object along with a driver class that tests the methods and constructor of that class. Their class description for the object they make must have at least 4 instance variables, a constructor that initializes values to those instance variables, and, it should have at least 2 additional methods that do different types of things. Creativity is encouraged!
IMPROVED BINARY NUMBER TRICK PROGRAM: Students will create a method that accomplishes the same algorithm as one previously called multiple times in the Binary Number Trick program from Unit 2, but is written one time as a method and called upon with a parameter that is a String of values.
NAME ECHO PROGRAM:

Students will write a program that asks for the user's name and then writes it back with the first name as entered, and the second name all in capital letters. Assume that there are two names, and that they are separated by a single space character. They should use the trim() method to remove possible leading spaces.
PLAYING CARD OBJECTS:
Students will write code that creates playing card objects. This program must:
Include the use of getter and setter methods for private variables.
Utilize two different constructors, one for a random card, and one for a specified card based on a parameter for rank and one for suit.
A method that prints out the name of the card.
A method that determines if two card match in rank
A method that determines if two card match in suit
OBJECT INCEPTION!:
Students will improve on a previous assignment with a class that contains objects of the first type you created (for example a car dealership object that contains car objects). They may use any previous work from this unit.

Teacher Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 5 - Arrays

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-DA-08 Collect data using computational tools and transform the data to make it more useful and reliable. (P6.3)
3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. (P4.1)

Vision of A Learner Attributes:
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA4(9-12): I can create opportunities to extend my learning by remaining open-minded in any situation.

Understandings: Students will understand that...
- An Array is a data collection tool that allows for organization of real world lists such as rosters, statistics, testing results, etc.
- Variables can be referenced through the name of a collection object and the index of that object more easily than through the use of numerous variable names.
- A multi-dimensional array is structured as an array of an array.

Essential Questions:
- How can we create visual representations in a Java program?
- Why is using an array better than using a large number of variables?
- What types of data can be stored in an array?

Students will know...
- An array is an object.
- Each cell of a numeric array is initialized to zero; each cell of an array of references is initialized to null.
- A statement of the form arrayB=arrayA will only copy the object reference in arrayA to arrayB, it will not change the contents of arrays.
- Because an array index starts at 0, but counting loops usually start at 1, there is a high likelihood of error without careful attention to detail.
- Arrays can contain string references inside cells.

Students will be able to...
- Initialize, adjust, and read values from an array in a variety of Java programs.
- Use methods of the ArrayOps class to change arrays.
- Construct two dimensional arrays for use in Java programs.
- Utilize the StringBuffer class to modify strings in Java programs.

Key Vocabulary: Array, Cells, Index, Subscript, Element, Subscripted Variable, Length of an array, null, initializer list, one dimensional array, algorithm, off-by-one error, ArrayOps class, findMax(), call by values, print(), copy(), String[] args, command line arguments, parseInt(), search(),

two dimensional array, row, column, bounds checking, charAt(), StringBuffer class

Assessment Evidence

Performance Tasks:
Unit 5 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts.
The options include programs to:

- Generate a 2-d array of specific size such that each cell is the product of its indices
- Replicate the game “Codebreaker”
- Edit a list of strings from a database of NFL data
- Create and solve Word Search puzzles

Other Evidence:
Unit 5 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

GRADE LIST PROGRAM:
Students will write a program that stores grades in a 1-d array. Scores can be accessed, modified, or deleted based on input from the user. As an extension, students may try to write an algorithm to figure out the average of the grades at any point.
BASEBALL LINEUP OBJECTS PROGRAM:
Students will write a class that creates Baseball Lineup objects. These objects have 1 instance variables: a String array of size 9. The constructor should initialize the String array so that it holds the names of 9 baseball players (Real or fake, your choice). Multiple methods should be included in this class.
One method is public void swapTwoPlayers(int index1, int index2). This method should swap the locations of two players in the lineup.
A second method is public void nextUp( ). This method will take the batter in position 1(index 0) and move them to the end of the lineup (index 8) after sliding every other player closer to index 0 in the line up. This method MUST do so by using the swapTwoPlayers method to, one at a time, move player 1 towards the back of the lineup.
A third method is public int whereInLineup(String name). This method looks for the parameter name in the array and returns the POSITION that the player was found at. Remember that position is not the same as index. The player in position 1 is at index 0, the player in position 2 is at index 1, and so on.
The final method is public void randomizeOrder( ). This method should randomly rearrange the players in the lineup(array). It MUST do so by

using the swapTwoPlayers method by picking random ints for index 1 and index 2 some large number of times.
SEATING CHART ACTIVITY:
Students will write a program that makes seatingChart objects. These objects contain an instance variable for a 2d String Array. There should be two constructors. One that populates the seating chart with names passed in as parameters and one that doesn't. One method, moveToSpot would move a student in the seating chart to a specified spot(don't allow if a student is already there). Another method switchItUp() randomizes the location of all students. Other methods include transferOut() and transferIn(). Spaces with no student are set to null.

Teacher Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 6 - Advanced Object Oriented Programming

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
3A-AP-14 Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables. (P4.1)
3A-AP-17 Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects. (P3.2)
3A-AP-18 Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs. (P5.2)

Vision of A Learner Attributes:
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.
AA2(9-12): I can assess my past successes and mistakes to change my approach.

Understandings: Students will understand that...
- Inheritance allows for related representations of objects to be created without repetition of common traits or procedures.
- Abstraction and Polymorphism allow us to have, analyze, and manipulate generalized collections of data

Essential Questions:
- How does abstraction improve the functionality, and readability, or a program?
- What types of real-world scenarios can be represented through the use of a hierarchy (and by extension, parent and child classes)?
- How are abstract classes and interfaces similar and how are they different?

Students will know...
- A child class inherits all the instance variables, base constructor, and methods of its parent class.
- Invoking the constructor of a super class allows for modifications to the original constructor.
- An abstract class cannot be instantiated, but it serves as a parent to multiple classes.
- All abstract methods must be well-defined in a child class.
- The use of an interface allows for multiple inheritance.
- Polymorphism allows us to repeat a procedure on several related, but different, classes.

Students will be able to...
- Create more complex programs that utilize multiple classes, including abstract classes and interfaces to model the relationship between different types of objects.
- Utilize the Comparable class to make decisions about strings of text.
- Implement an ArrayList for any data type and add, modify, and remove elements as needed.

• ArrayLists are Arrays whose size can be modified.	

Key Vocabulary: Inheritance, Parent Class, Child Class, Single Inheritance, Multiple Inheritance, Hierarchy, Root, Extends, Super( ), Override, Abstract Class, Abstract Method, Polymorphism, Descendants, Signature, Interface, Implements, Type Cast, Comparable, CompareTo( ), Collating Sequence, Binary Search, ArrayList, List, Iterator, Enhanced For Loop, Wrapper Class, Autoboxing, Null

Assessment Evidence

Performance Tasks:
Unit 6 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts.
The options include programs to:

- Compare two or more diamonds based on clarity and color grade
- Represent various 3-d solids and find volume and surface area of them
- Order and edit a list of strings by length
- Represent zoos and animals with abstract objects

Other Evidence:
Unit 6 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

EXAMPLE OF INHERITANCE PROGRAM:
Students will demonstrate inheritance by making a project in BlueJ that has a parent class and several child classes. The child classes should (meaningfully) inherit commonalities in instance variables and methods from its parent. Additionally, each child class should have at least one instance variable or a method that is unique to it. Each child class should use the super ('parameters') command in its constructor to initialize all the instance variables it inherited before initializing any instance variables specific to it. Finally, in at least one of the child classes a method inherited from the parent should be overwritten.
POLYGON ABSTRACT CLASS OR INTERFACE:
The class will be split into two halves. One half of the class will model the hierarchy of quadrilaterals with an Abstract class called “Quads” and the other half using an Interface of the same name. Both are possible but must be implemented in different ways. Each subclass spells out the different traits of various quadrilaterals(Kite, Trapezoid, Isosceles Trapezoids, Parallelograms, Rectangles, Rhombi, and Squares) and multiple

levels of subclass may be used. A class discussion will follow with an analysis of the difference between the two approaches.
TRIOS ACTIVITY:
Students will write a class of objects called "UserGroups". A UserGroup object has three private string instance variables that represent online usernames. The Constructor will initialize these instance variables with three string parameters. The class should have two methods. The first method called "anyDuplicates( )" will use the compareTo( ) method to check if any of the two usernames are the same and return a boolean true if there are any duplicates and false otherwise. The second method called "userNamesInOrder( )" will use a technique like in "Trios" to return a String array of size 3 with the names in order. Use compareTo( ) to order the usernames by Lexicographical order as shown on chortle.ccsu.edu/java 5 in chapter 53B.
BINGO CALLER PROGRAM:
Students will write a program that randomly generates values from 1-75 and displays them on the screen to the user, preceded by the appropriate letter (B for 1-15, I for 16-30, etc.) for as long as the user requests another number. An ArrayList will be used, either of size 75 and for booleans, or an unsorted ArrayList of values called, to ensure that no repeats are called.

Teacher Resources:
http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 7 - Recursion

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-AP-17 Systematically test and refine programs using a range of test cases. (P6.1)
3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. (P5.2)

Vision of A Learner Attributes:
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Understandings: Students will understand that...
- Recursion is the process of breaking down a complex procedure into a repeated behavior and a base case. For example, to find a word in the dictionary, repeatedly do the following:
  -If the word is the target word, read it's definition (base case)
  -If it is not the word, move forward or backwards in the dictionary to get closer to the word.

Essential Questions:
- What are the key components of any recursive process?
- What are some advantages and disadvantages to using recursion?

Students will know...
- Recursion involves a base case, which describes the situation where recursion concludes, a recursive rule, and typically, some return statement to connect instances of the recursive process.
- Recursion can solve the same problems that a loop can, however, usually in a more efficient way.

Students will be able to...
- Write a recursive method that can perform the same process as any loop.
- Identify the result of a recursive process through a dynamic analysis of each iterative step in the process.

Key Vocabulary: Recursion, Base Case, Triangular Numbers, Dynamic View, Return, Factorial, Numerical Analysis, Iterative Implementation, Hailstone Numbers, Newton's Method

Assessment Evidence

Performance Tasks:
Unit 7 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a

Other Evidence:
Unit 7 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what

program for their choice of 3 of 4 prompts. The options include programs to: ● Sum a list of numbers ● Verify if a given string is a palindrome ● Determine the amount of a radioactive material left after some number of half lives ● Counts the number of various characters in a string of text	snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

RECURSION GROUP DISCUSSION:
Students will cement their understanding of the recursive process, before putting it to use in code, by answering the following prompts: (choose 3 of the 9 then discuss in groups)
Describe with recursion how to stir a can of paint until the paint is thoroughly blended.
Describe with recursion how to read a 1000 page book.
Describe with recursion how to sit through a boring 60 minute lecture.
Describe with recursion how to dig a ditch that is big enough to bury an elephant.
Describe with recursion how to wax your car until every surface gleams.
Describe with recursion how to visit every exhibit in a zoo.
Describe with recursion how to drink all the fluid in a liter bottle of grape soda.
Describe with recursion how to play one hole of golf (how to move the ball from the tee to the hole).
Describe with recursion how to work every exercise in a list of exercises.
NEWTON'S METHOD PROGRAM:
Students will replicate the process of Newton's Method by writing code that takes in the coefficients of a polynomial and, through recursion, finds estimates to 5 decimal places of the roots of the polynomial.

Teacher Resources:

http://chortle.ccsu.edu/Java5/index.html#07
Java Runtime Environment
Java Development Kit

Unit 8 - Searching and Sorting

Desired Results - Goals, Transfer, Meaning, Acquisition

Established Goals:
2-AP-13 Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs. (P3.2)
3A-AP-13 Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests. (P5.2)

Vision of A Learner Attributes:
TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
TCC4(9-12): I can integrate my learning to adapt to experiences in the classroom, career and life.

Understandings: Students will understand that...
- There are dozens of various searching and sorting algorithms, each with distinct advantages and disadvantages.
- Almost every occupation requires the ability to search or sort data in a meaningful way.

Essential Questions:
- What makes a searching or sorting algorithm efficient?
- How do the gains in efficiency of a sorting algorithm balance against the added complexity to program it?
- Can all searching and sorting problems be solved with an algorithm?

Students will know...
- Linear Search is less efficient that Binary Search, but does not need a sorted list of elements to function.
- Bubble sort is the least efficient, but easiest to code, of all the common sorting algorithms.
- Insertion Sort and Selection Sort have various advantages/disadvantages each with unique best and worst case scenarios.
- Merge Sort is an efficient sorting method that utilizes a “divide and conquer” approach.

Students will be able to...
- Write the algorithm for various searching and sorting techniques.
- Evaluate the speed of an algorithm in terms of number of processes.
- Debate the advantages/disadvantages to various algorithms.

Key Vocabulary: Algorithm, Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Running Time, Best Case, Worst Case, Lexicographic Order, Linked List, Node, Pointer, Traversal

Assessment Evidence

Performance Tasks:	**Other Evidence:**
Unit 8 Test- Students will demonstrate their understanding of the content of this unit through an open-ended test requiring them to code a program for their choice of 3 of 4 prompts. The options include programs to: ● Sorts a list of data using Insertion Sort, but from the center out ● Compare the speed of various sorting methods ● Modify the standard Merge Sort algorithm to use 4 subdivisions instead of 2 ● Sort a randomly ordered deck of cards	Unit 8 Quiz - This quiz consists of a series of released Multiple Choice problems from AP CSA Exams, and requires students to assess what snippets of code do, debug code, or to evaluate different ways of accomplishing the same task.

Learning Plan

Students will demonstrate the following VoL Attributes:

TCC3(9-12): I can integrate relevant information to produce multiple valid solutions.
AA1(9-12): I can evaluate different approaches and justify the best pathway to success.

Through the completion of these Formative Assessments:

“IT’S A RACE 1” ACTIVITY:
Students will use both the linear search and binary search algorithms on a set of 1000 sorted integers to find a specific value. First, they will code algorithms for both techniques and, within each loop, they will establish a counter to determine the number of steps that each process took. The average result for all tests should show n steps for n integers and log(2)n steps for binary search.
REPEAT INTEGERS PROJECT:
Students will write a method that counts how many integers appear more than once in a sorted array of integers. In other words, if the count is zero, then no integer has appeared twice or more. If the count is one, then some integer has appeared more than once. If the count is N, then N different integers have appeared more than once. This can be done by scanning through the sorted array once and incrementing a duplicate counter every time an integer appears twice or more in a row. Of course, this only works if the array is sorted. It is much more work to find duplicates in an unsorted array.
INSERTION SORT ACTIVITY:
Students will write an insertion sort where the sorted sublist is on the right of the list. Sorting now works by taking the element just to the left of the sublist and inserting it where it belongs. A sublist of size R contains the R largest elements in the array, in sorted order.
“IT’S A RACE 2” ACTIVITY:
Students will use the merge sort algorithm on a set of data found on the internet that contains at least 10000 data values. They should attempt to run the same data through another simpler sorting algorithm and record the time(with the stopwatch feature of their phone) that the sorting requires in both cases.
ARRAYLIST REVISED PROGRAM:
Students will revisit the Bingo Caller Program from a previous unit. They should replace the code that handled the collection and organization of

called numbers in an arraylist, now with a linked list. This program provides extra functionality, and allows for easier adjustment to the list.
Teacher Resources: http://chortle.ccsu.edu/Java5/index.html#07 Java Runtime Environment Java Development Kit