AP Computer Science A Pacing Guide

Content Area: **CTE**
Course(s): **AP Computer Science A**
Time Period: **Sept-June**
Length: **180 Days**
Status: **Published**

AP Computer Science A Pacing Guide

Belleville Public Schools Unit Pacing Guide

Content Area: Business/Technology
Course(s): Computer Science A
Time Period: Full Year

Division of Units / Topics:

Unit Plan 1 Fundamentals of Programming - Primitive Types	25+ days
Unit Plan 2 Using Objects - Control Statements and Loops	25+ days

Unit Plan 3 Strings -Boolean Expressions and if statements -Iteration Standard Java programs include the String and Wrapper classes. 1. Students will manipulate Strings and use methods from the String class. 2. Students will implement the Wrapper class to represent primitive data as objects. 3. Students will implement the static methods of Wrapper classes to perform particular algorithms	25+ days
Unit Plan 4 Arrays Arrays are data structures used to group homogeneous data to facilitate data manipulation and organization. 1. Students will implement arrays of primitive values as well as of arrays of object values. 2. Students will implement and use a multi-dimensional array. 3. Students will design nested for loops to populate data in a multi-dimensional array.	25+ days
Unit Plan 5 Array List / 2D Array Arrays are data structures used to group homogeneous data to facilitate data manipulation and organization. 1. Students will implement arrays of primitive values as well as of arrays of object values. 2. Students will implement and use a multi-dimensional array. 3. Students will design nested for loops to populate data in a multi-dimensional array.	25+ days
Unit Plan 6 Inheritance Inheritance is a technique for organizing and creating hierarchies of classes used to form polymorphic references. 1. Students will	25+ days

construct a hierarchy of classes that inherit each other's properties. 2. Students will demonstrate the use of the keyword super to facilitate the construction of methods and constructors in the hierarchy. 3. Students will demonstrate the difference between 'is a' relationship and 'has a' relationship. 4. Students will demonstrate what overridden methods are. 5. Students will demonstrate the use of an interface.	
Unit Plan 7 Recursion Recursion is a necessary and powerful programming technique used to solve specific problems. 1. Students will apply the underlying ideas of recursion. 2. Students will understand that each recursive algorithm needs a base case to end recursion. 3. Students will compose programs that incorporate recursion to solve problems.	25+ days

Computer science involves problem-solving, hardware, and algorithms that help people utilize computers and incorporate multiple perspectives to address real-world problems in contemporary life.

As the study of computer science continues to evolve, the careful design of the AP Computer Science A course strives to engage a diverse student population, including female and underrepresented students, by allowing them to discover the power of computer science through rewarding yet challenging concepts.

A well-designed, modern AP Computer Science A course that includes opportunities for students to collaborate to solve problems that interest them, as well as ones that use authentic data sources, can help address traditional issues of equity and access. Such a course can broaden participation in computing while providing a strong and engaging introduction to fundamental areas of the discipline.

The AP Computer Science A course reflects what computer science teachers, professors, and researchers have indicated are the main goals of an introductory, college-level computer science programming course: Program Design and Algorithm Development— Determine required code segments to produce a given output. Code Logic—Determine the output, value, or result of given program code given initial values. Code Implementation—Write and implement program code. Code Testing—Analyze program code for correctness, equivalence, and errors. Documentation—Describe the behavior and conditions that produce the specified results in a program. Ethical Computing—Understand the ethical and social implications of

computer use. Students practice the computer science skills of designing, developing, and analyzing their own programs to address real-world problems or pursue a passion

Big Idea 1: Modularity

Big Idea 2: Variables

Big Idea 3: Control