Course Title: AP Computer Science A

Course #: 3467-3468

Course Description:
AP Computer Science A is equivalent to a first-semester, college-level course in computer science. The 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. The course emphasizes both object-oriented and imperative problem solving and design using Java language. These techniques represent proven approaches for developing solutions that can scale up from small, simple problems to large, complex problems. The AP Computer Science A course curriculum is compatible with many CS1 courses in colleges and universities. Students are strongly encouraged to take the AP Exam in May.

UC/CSU Approval: “g” approved

Grade Level: 11-12

Estimated Homework Per Week: 1hr/wk
This is a general guideline for planning and scheduling purposes. A student’s ability level may affect actual preparation time needed.

Prerequisite: Completion of Pre-Calculus or AP Computer Science Principles with a grade of “B” or higher

Recommended Prerequisite Skills: Logic, Problem Solving

Course Grade Categories:
• Assessments: 35%
• Final Assessment: 15%
• Homework/Classwork: 50%
Major Assessments/Units/Topics:
*Assessments per unit: 1-2 quizzes, 1 Unit Test, 3-4 Programming Projects

Unit 1: Introduction to Computing:
- Computer Processing
- Hardware Components
- Java Programming Language
- Object-Oriented Programming

Unit 2: Data & Expressions:
- Character Strings
- Variables and Primitive Data Types
- Expressions, Data Conversion
- Interactive Programs
- Graphics

Unit 3: Classes & Objects:
- Creating Objects
- Packages String
- Random
- Math Classes
- Formatting Output
- Enumerated Types
- Wrapper Classes
- Components & Containers
- Nested Panels
- Images
- Writing Classes & Objects
- Anatomy of a Class
- Encapsulation
- Anatomy of a Method
- Constructors
- Graphical Objects
- Graphical User Interfaces (GUI)
- Buttons & Text Fields

Unit 4: Conditionals & Loops:
- Simple Operations
  - Boolean Expressions and If-Statements
  - The While Statement and Iterators
  - The ArrayList Class
  - Event Sources
  - Check Boxes and Radio Buttons
*Complex Operations*
  - The Conditional Operator,
  - Statements: do, for, switch
  - Drawing w/ Loops and Conditionals
  - Dialog Boxes

**Unit 5: Object-Oriented Design:**
- Software Development Activities
- Identifying Classes and Objects
- Class Relationships
- Interfaces
- Method Design and Overloading
- Testing
- GUI Design

**Unit 6: Arrays:**
- Array Elements
- Declaring and Using Arrays
- Arrays of Objects
- Variable Length Parameter Lists
- Two-Dimensional Arrays
- Polygons and Polylines
- Mouse and Key Events

**Unit 7: Inheritance:**
- Creating Subclasses
- Overriding Methods
- Class Hierarchies
- Designing for Inheritance
- Extending Adapter Classes
- Timer Class

**Unit 8: Polymorphism**
- Late Binding via Inheritance and Interfaces
- Sorting and Searching
- Event Processing
- File and Color Choosers
Unit 9: Exceptions
- Exception Handling
- Uncaught Exceptions
- The try-catch Statement
- Exception Propagation
- I/O Exceptions
- Tool Tips and Mnemonics
- Combo Boxes
- Scroll Panes and Split Panes

Unit 10: Recursion
- Recursive Thinking & Programming
- Using Recursion
- Recursion in Graphics

Unit 11: Collections
- Collections and Data Structures
- Dynamic Representations
- Linear Data Structures
- Non-Linear Data Structures
- Java Collections API

Unit 12: AP Prep Labs
- Magpie Lab
  - Strings and AI
- Elevens Lab
  - Classes and Objects
- Picture Lab
  - Arrays, Interfaces, Inheritance