

OHHS Career and Technology Computer Science Course Syllabus



Course Name: AP Computer Science A
Course Code: CTO55, CTO56, CTO57
Prerequisite: Geometry, 11th, and 12th
Length: Year
Course Fees: None

COURSE DESCRIPTION:

This course is designed to introduce students to computer science topics of problem solving, design strategies and methodologies, data structures, algorithms, analysis of potential solutions, and the social and ethical issues related to computer use. The Java programming language is used throughout the course to achieve the above.

COURSE RESOURCES/TEXTBOOK:

- Litvin, Maria, and Gary Litvin. *Java Methods: Object-Oriented Programming and Data Structures*, 3rd AP Edition, Andover, Mass.: Skylight Publishing, 2015.
- Litvin, Maria, and Gary Litvin. *Be Prepared for the AP Computer Science Exam in Java*, 6th Edition, Andover, Mass.: Skylight Publishing, 2014.
- The College Board's Magpie, Picture, and Elevens Labs Student Guides.
- CodingBat: <http://codingbat.com/java>
- CodeHS: <http://codehs.com>
- Runestone Academy: <https://runestone.academy/>
- Litvin, Maria, and Gary Litvin. *250 Multiple-Choice Computer Science Questions in Java*. Andover, Mass.: Skylight Publishing, 2008.

GRADING SCALE:

- 60% Exams 30% Daily Work (Assignments/Labs) 10% Professionalism
- Points will be given and a grade based on your percentage of points on assignments, labs, quizzes, and tests. The grading scale reflects the rigor of an AP course and has been agreed upon by all AP teachers.

90-100% (or 5 on AP Exam)	A	77-82.9% (or 3 on AP Exam)	B	67-72.9% (or 2 on AP Exam)	C
85-89.9% (or 4 on AP Exam)	A-	75-76.9%	B-	65-66.9%	C-
83-84.9%	B+	73-74.9%	C+	55-64.9%	D
				below 55%	F

Exams/Assessments:

- Quizzes, Chapter Tests, Final

Daily Work:

- Assignments, Labs

AP COMPUTER SCIENCE A

Professionalism:

- You will typically earn 2 points a day for coming to class and working in a professional, cooperative, and productive manner. If you are absent write me an email or note the next day explaining your absence. Extra points are available for exemplary work habits.
- Be on time
- Be in uniform, dressed appropriately
- Be on task – working on appropriate tasks
- Respecting others with words and actions – builds others up!
- Obeying class rules

COURSE DESIGN:

The proposed syllabus is for a three trimester course, assuming 30 weeks are available prior to the AP exam. The course meets for five 67-minute class periods per week. The course includes a number of individual programming projects assigned for one week each. The time after the AP CS Exam is devoted to a team project and enrichment activities.

The course is based on numerous problem solving exercises, labs, and case studies, which require students to design and implement Java classes. The course requires 40-50 hours of hands-on work in a computer lab.

COURSE OBJECTIVES:

- Understand and apply the main principles of object-oriented software design and programming: classes and objects, constructors, methods, instance and static variables, inheritance, class hierarchies, and polymorphism
- Learn to code fluently in Java in a well-structured fashion and in good style; learn to pay attention to code clarity and documentation
- Learn to use Java library packages and classes within the scope of the AP Java subset
- Understand the concept of an algorithm; implement algorithms in Java using conditional and iterative control structures and recursion
- Learn to select appropriate algorithms and data structures to solve a given problem
- Compare efficiency of alternative solutions to a given problem
- Learn common searching and sorting algorithms: Sequential Search and Binary Search; Selection Sort, Insertion Sort, and Mergesort
- Understand one- and two-dimensional arrays, the List interface, and the ArrayList class, and use them appropriately in programming projects
- Acquire skills in designing object-oriented software solutions to problems from various application areas
- Discuss ethical and social issues related to the use of computers
- Prepare for the AP Computer Science A exam; meet all of the curricular requirements defined by the College Board for this course.

AP COMPUTER SCIENCE A

COURSE OUTLINE:

Chapter numbers for readings and exercises refer to Java Methods, 3rd AP Edition. The labs, case studies, and projects proposed below come from Java Methods. *Assignments & Labs for a given chapter are due on the exam date.*

Unit 1: An introduction to computers and software engineering (2 weeks)

1. Hardware, software and the Internet (Week 1; duration 1 week) Ch. 1 Exam - Wed. Sept. 11

Elements of a computer system. How information is represented in computer memory. Binary and hex number systems and ASCII / Unicode. An introduction to the Internet.

Reading and exercises: Chapter 1 {1-17,19,20}.

Lab: Find and explore the home pages of some Internet and World Wide Web pioneers.

Lab: *Picture Lab*, Activities 1 and 2.

2. An introduction to software engineering (Week 2; duration 1 week) Ch. 2 Exam - Wed. Sept. 18

Getting familiar with the software development process. Compilers and interpreters. JDK tools (*javac*, *java*, *javadoc*). Running a Java program in a command-line environment (optional). Using an IDE. Java classes and source files. A brief introduction to OOP. Software engineer's Code of Ethics.

Reading and exercises: Chapter 2 {1-12, ≥ 3 POYCs 14-18} and Section 28.3

Lab: Compile and run simple programs (Hello World, Greetings) using command-line JDK tools or an IDE (Section 2.4).

Lab: Compile and run simple GUI applications (Section 2.6).

Unit 2: Syntax and objects (3 weeks)

3. Java syntax and style (Week 3; duration 2 weeks)

Ch. 3 Exam - Tue. Oct. 1

Syntax and style in a programming language. Comments. Reserved words and programmer-defined names. Statements, braces, blocks, indentation. Syntax errors, run-time errors, logic errors.

Reading and exercises: Chapter 3 { ≥ 8 POYCs}; Appendix A.

Lab: Pounds to Kilograms converter

Lab: Red Cross

Lab: Smiley Face

Lab: 6x6 varicolored checker board

Lab: Correcting syntax errors and a logic error as an “adventure game” (Section 3.7).

codehs.com: Introduction To Programming In Java With Karel The Dog

AP COMPUTER SCIENCE A

4. A first look at objects and classes (Weeks 4-5; duration 2 weeks) **Ch. 4 Exam- Tue. Oct. 15**

Classes and objects. Classes and source files. CRC cards. Library classes and packages. The `import` statement. A first look at fields, constructors, and methods of a class. Inheritance.

Reading and exercises: Chapter 4 { ≥ 9 POYCs} and *Elevens* Lab Student Guide, Activity 1

Lab: Design and implement `Circle` and `Cylinder` classes (Exercise 8, p. 92).

Case study: `BalloonDraw` (Section 4.2).

Case study and lab: `Balloons of All Kinds` (extend the `Balloon` class, coding constructors and overriding methods (Section 4.6).

Lab: *Elevens*, Activity 1.

Unit 3: Arithmetic, logic, and control statements (7 weeks)

Ch. 5 Exam -Thur. Oct. 31

5. Data types, variables, and arithmetic (Weeks 6-7; duration 2 weeks)

The concepts of a variable and a data type. Declarations of variables. Fields vs. local variables. The primitive data types: `int`, `double` and `char`. Literal and symbolic constants. Initialization of variables. Scope of variables. Arithmetic expressions. Data types in arithmetic expressions. The cast operator. The compound assignment (`+=`, etc.) and increment and decrement operators (`++`, `--`). Converting numbers and objects into strings. Math methods (`abs`, `sqrt`, `pow`, `random`).

Reading and exercises: Chapter 5 { ≥ 10 POYCs}

Lab: `JavaMethods` exam question #26

Lab: `JavaMethods` book question #18 .

Lab: `Rainbow` (Exercise 27, p. 130).

codehs.com: Basic Java

6. The `if-else` statement (Weeks 8-9; duration 2 weeks)

Ch. 6 Exam - Tue. Nov. 19

The `if-else` statement. Boolean expressions, the `boolean` data type, `true` and `false` values. Relational and logical operators. De Morgan's laws. Short-circuit evaluation. Nested `if-else` and `if-else-if`. *Case Study: Craps*. Elements of object-oriented design in *Craps*. The `switch` statement. `enum` data types.

Reading and exercises: Chapter 6 { ≥ 10 POYCs}.

DeMorgan's Law poster

Lab: The `Die` and `CrapsGame` classes for *Craps*: fill in the blanks and test in isolation (Section 6.9).

Lab: Finishing and testing the *Craps* program (Section 6.12).

codingbat.com *Logic-1* and *Logic-2*.

AP COMPUTER SCIENCE A

7. Algorithms and iterations (Weeks 10-12; duration 3 weeks) **Ch. 7 Exam - Tue. Dec. 10**

The concept of an algorithm. Properties of algorithms. Iterations. `while`, `for`, and `do-while` loops. `break` and `return` in loops. Nested loops. Euclid's GCF algorithm.

Reading and exercises: Chapter 7 { ≥ 8 POYCs }.

Lab: EvenProducts

Lab: PrimeList

Lab: PrimeSum

Lab: AddSquares Multiply

Lab: AddSquares Addition

Lab: Print stars using iterations (Exercise 20, p. 204).

Case study and lab: Euclid's GCF algorithm (Section 7.7 and Exercise 26 on p. 206).

Lab: Perfect Numbers (Section 7.8).

Interlude: Ethical and social implications of computer use (Week 13, 1 week)

Student papers, presentations, and debates on ethical and social issues related to the use of computers and the Internet.

Reading: Sections 28.3 - 28.5; current news and commentary in the online media.

Unit 4: Strings and arrays (4 weeks)

8. Strings (Week 14; duration 1 week)

Ch. 8 Exam - Thur. Jan. 9

String objects. Literal strings. Immutability. String methods. Converting strings into numbers and numbers into strings. The `Character` class and its methods.

Reading and exercises: Chapter 8: `compareTo()`, `.equals()`, `replace()` { ≥ 8 POYCs }.

Lab: `charAtCode` - pencil & paper

Lab: `charAtCode` - computerized

CodingBat: String1, String2,

CodingBat Bonus: String 3

codehs.com: Methods

Optional:

Lab: Lipographs pg 224 - 226 of JavaMethods

Lab: Magpie Activities 1 & 2

AP COMPUTER SCIENCE A

9a. One-dimensional arrays (Weeks 15-16; duration 1.5 weeks)

One-dimensional arrays. Arrays as objects. Declaring and initializing. Indices. Length. `IndexOutOfBoundsException`. Traversals and the "for-each" loop. Inserting and removing elements.

Reading and exercises: Chapter 9 { ≥ 8 POYCs }.

Lab: studentList Array - create a class title Students that contains an array called studentList that contains the first names of ten students in this class. Number the students 1 through 10. Print out the ten names from the array on separate lines using a for loop. Declare the array explicitly as described on page 238.

Lab: For-Each: Alter studentList array so that a for each loop is used to traverse the array and print out the student names

Lab: ArrayPractice - practice creating new arrays - paper handout. Create mlkArray as shown and add For loop as directed to print message.

Lab: MaxNumber - write a findMax method that traverses the array and returns the maximum value. - attached

Lab: InsertArrayElement - insert an element in an array, moving the affected to the right - attached

Lab: Fortune Teller pg 241. Copy the files from EasyClasses into the fortune folder. Follow the directions on pg 241 to include a minimum of 5 fortunes.

Lab: Magpie, Activity 5.

Lab: Chomp Lab - use printout.

Lab: Past free-response questions on arrays.

Case study and lab: *The Sieve of Eratosthenes* (Section 9.8).

`codingbat.com`: Minimum: Array 1: 20 problems, Array 2: 20 problems

9b. Two-dimensional arrays (Wks 16-17; duration 1.5 wks) Ch. 9 Exam: Tue Jan 28

Declaring and initializing two-dimensional arrays. Accessing the number of rows and columns. Traversals and nested "for-each" loops.

Reading and exercises: Chapter 9 {see above}

Lab: Past free-response questions on arrays.

Lab: Chomp (Section 9.5).

Lab: 2D Arrays worksheet - Alphabet class

Lab: 2D_PowersArray - use 2d arrays to create 3 columns of numbers with powers of 1, 2, & 3 respectively - attached

`codingbat.com`: Array 3: 3 problems

Unit 5: Classes and class hierarchies (6 weeks)

10. Details of defining classes and using objects (Weeks 18-19; 2 weeks) Ch.10 Exam - Tue. Feb. 11.

Public and private fields and methods. Constructors and the `new` operator. References to objects. Calling methods and accessing fields. Passing parameters to constructors and methods. `return` statement. Overloaded methods. Static variables and methods.

Reading and exercises: Chapter 10 { ≥ 10 POYCs }.

Case study: the `Fraction` class (Sections 10.1 - 10.8).

Case study and lab: *Snack Bar* (Section 10.9).

Lab: Snack Bar part 2 pg 306 & 307 - replace `SnackBar.java` with the file from the `SnackBar-Part2` folder. Update the `Vendor` class so that Total Sales will be shown as per directions on page 307.

`codehs.com`: Classes and Object-Oriented Programming (4.1.1 - 4.11.7)

11. ArrayList (Weeks 20-21; duration 2 weeks)

Ch. 11 Exam - Thur. Feb. 27

ArrayList structure. The List interface. ArrayList's constructors and methods. Pitfalls. ArrayList vs. built-in arrays.

Reading and exercises: Chapter 11 { ≥ 5 POYCs }.

Be Prepared: Section 2.5.

Lab: pg 323 *Shuffler* (Section 11.4).

Lab: Creating an index for a document — using ArrayList and writing a subclass of ArrayList (Section 11.6).

Lab: Past AP free-response questions on ArrayList.

Lab: *ECG Analysis* (*Be Prepared*, Practice Exam 3, Question 1).

codehs.com: Data Structures sections 1 - 8 (through Hashmaps).

12. Class hierarchies, abstract classes, and interfaces (Wks 22-23; 2 wks) Ch.12 Exam - Thur. Mar. 12

Class hierarchies. Abstract classes. Invoking superclass's constructors and calling superclass's methods. Polymorphism. Interfaces.

Reading and exercises: Chapter 12 { ≥ 12 POYCs }.

Lab: Abstract Class Lab

Lab: Interface Lab

Lab: Quadrilateral: Inheritance, Abstract Classes & Methods, Polymorphism

Lab: 2015 AP Exam free response question 4 using interfaces.

CodeHS: Classes & Object Oriented Programming sections 12 (4.12.1) through 15 (4.16.1) - includes quiz: Inheritance, Class Design and Abstract Classes, Polymorphism, Interfaces.

Unit 6: Recursion, searching and sorting (4 weeks)

13. Recursion (Week 24; duration 1 week)

Ch. 13 Exam - Thur. Mar. 26

Recursive methods. Base case. Understanding and debugging recursive methods. When not to use recursion.

Reading and exercises: Chapter 13 { ≥ 10 POYCs } and Sections 23.3 - 23.5.

Lab: Chapter 13 exercises (for example, 6, 10 pp. 372 - 373).

CodingBat: ≥ 24 recursive problems.

CodeHS: Algorithms & Recursion: 6. Advance Recursion Sections 6.6.1 - 6.6.5

AP COMPUTER SCIENCE A

14. Searching and sorting. Introduction to analysis of algorithms. (Weeks 25-27; duration 3 weeks)

Ch. 14 Exam - Thur. Apr. 16.

Comparing objects. The `equals` method and the `Comparable` interface. Sequential and Binary Search. The number of comparisons required in Sequential and Binary Search. Selection Sort, Insertion Sort, and Mergesort. Comparison of efficiency of "quadratic" sorting algorithms (Selection Sort and Insertion Sort) vs. Mergesort.

Reading and exercises: Chapter 14 { ≥ 8 POYCs}.

Labs: Linear & Binary Search (Strings & Doubles); Selection Sort, Insertion Sort, Merge Sort

Lab: *Keeping Things in Order* (Section 14.4).

Lab Demo: *Benchmarks* (Section 14.9) — compares efficiency of several sorting algorithms.

codehs.com: Algorithms and Recursion 6.1.1 - 6.8.1

Unit 7: Review (3 weeks)

15. Review and practice for the AP exam(Wks 28-30; duration 3 wks) **AP Exam - Fri. May 8 @**

noon

Quick reference (library classes and methods). Past multiple-choice and free-response questions.

Reading: *Be Prepared* Chapters 1-5; *Be Prepared* Chapter 6 (past free-response questions and solutions), *Be Prepared* practice exams 1-5, *250 Multiple-Choice Computer Science Questions in Java*.

codehs.com: AP Test Practice, Final Project

Unit 8: Enrichment (optional, duration varies)

16. Streams and files

Text and binary files. Streams vs. random-access files. Java I/O package. The `Scanner` class. Checked exceptions.

Reading and exercises: Chapter 15.

Lab: *Choosing Words* (Section 15.5).

Lab: Exercises and projects from exercises and the Test Package for Chapter 15.

17. Graphics and GUI

Computer graphics concepts. The Java `Graphics` class. GUI components and their events. Layouts. Handling mouse and keyboard events and images.

Reading and exercises: Chapters 16, 17, 18.

Lab: *Pieces of the Puzzle* (Section 16.7).

Programming project: *Rambles* (Section 17.6).

Lab: *Slide Show* (Section 18.7).

AP COMPUTER SCIENCE A

Unit 9: After the AP Exam (Duration varies)

Projects that demonstrate creative computer use.

Reading and exercises: Java Methods Chapter 28, “Computing in Context: Creative, Responsible, and Ethical Computer Use”, Section 28.2.

Other suggested activities: a team project to implement a game (for example, the Game of SET, <http://www.skylit.com/projects/> or the *Elevens* lab); or a potentially useful project for the school.

OHHS PLAGIARISM POLICY:

- Any student, who knowingly turns in any work that has been done by someone other than himself or herself, and fraudulently represents it as his/her own, shall be considered to have cheated. Cheating also includes: aiding someone else in cheating, the use or preparation of written, pictorial, or other materials not authorized by the instructor during a test or assignment, the use of testing materials obtained previous to the test date, or plagiarism of any kind. Students found cheating will also be subject to an office referral, which could result in a suspension. As an effort based school, students who have plagiarized/cheated must still demonstrate their learning. Thus, students will be given the opportunity to complete the work. This will fall under the “Late Work Policy” of the teacher where total possible points may be reduced

AP COMPUTER SCIENCE A

RE-DO POLICY:

OHHS is an effort based school where we believe all students can learn. We also know that students learn at different rates. Thus, we are implementing a re-do policy that recognizes the needs of individual learners.

- 100% Re-do
 - Assessments (with exception of the Final semester exam)
 - Key Assignments: Evidence that demonstrates mastery of a standard
- Teachers may:
 - require completion of supporting assignments prior to re-do opportunities for students to show they can meet the standard.
 - *All unit assignments must be completed and corrected by the student prior to retaking a unit exam.*
 - require students to complete re-do of work outside of class time.
 - *All make up work and re-dos/test retakes will be done during Tutoring/Extra Help times listed below.*
 - limit the number of re-do opportunities for each assignment.
 - *Students will have 1 re-do/retake opportunity for each test/project.*
 - assign zero grades when students choose not to do key assignments and/or assessments to standard.
 - have different requirements for late work.
 - *See Late Work Policy below.*
- All opportunities for re-do work will come with a deadline as assigned by the teacher.
 - *Students will have until the subsequent test is given to complete re-do work or to retake a test.*

LATE WORK POLICY:

- All assignments/labs can be completed up until the day of the subsequent/following unit exam (with the exception of codehs.com work). After that, no credit will be given.

TUTORING AND EXTRA HELP

- The computer classroom, room A216, will be open at 7:15 in the morning before school and it will stay open until at least 2:45 pm after school. Students are welcome to come in and work on assignments/labs during those hours.
- Students needing extra help can come in after school Tuesdays and Wednesdays.
 - Students unable to come in after school can make an appointment for before school or other afternoon if available.
- The computer classroom will be closed Mondays after school because of teacher meetings.

Instructor:

Bill Rodeheffer

Telephone: 360-279-5761

e-mail: brodeheffer@ohsd.net