

Equinox

Course Title: AP Computer Science A

Course Description

JAVA touches the life of almost every individual. Developed in 1995, this computer programming language is used in industries ranging from retail to finance to medicine. Students learn to program in JAVA using keywords, operators, and data types to develop solutions to problems, and subsequently to code and compile programs, as well as to compose command-line programs, basic graphics, and simple games. Students do not need prior experience with JAVA to take this course, but should have previous programming or computer language experience. This course prepares students for the AP Computer Science Exam.

Essential Questions

- What is method decomposition and how can it be used to simplify and test out solutions?
- How does object oriented design differ from a static-method driving solution?
- What are the three essential components of any Java class?

Outcomes

Upon successful completion of this course, students will:

- Know JAVA Syntax, how to compile JAVA programs using an IDE, and know the basics structure object-oriented design and algorithm design.
- Understand how to use control flow, classes, inheritance, and interfaces to write efficient code that breaks complex problems into different components that are solved by appropriate algorithms. Students will also understand how to test code and discover design flaws.
- Apply their knowledge to design JAVA programs both independently and in a group environment.
- Analyze requirements to design an efficient solution and to craft that solution using proper object oriented techniques.

Instructional Strategies

In order to meet the needs of students, the class will use flexible groups in order to have students assist each other and move at an appropriate pace. Additionally, all the assignments are tiered having both basic requirements and advanced options for students who wish to explore concepts in more detail or to explore concepts not directly covered in class.

Resources and Materials

- **Books**
 - a. Liang, Daniel Y: Introduction to Java Programming: Comprehensive Addition, Pearson, 2010, 8/e.
ISBN: 0132130807
- **Web sites**
 - a. AP Computer Science Central
http://apcentral.collegeboard.com/apc/public/courses/teachers_corner/4483.html
This website is the official College Board website for AP Computer Science A.
- **Materials**
 - a. Laptop computer capable of running a JAVA Compiler
 - b. IDE: BlueJ (www.bluej.org) or Eclipse (<http://eclipse.org/>)

Student Assessment

- **Pre-Assessment**

Practice AP Computer Science A exam.

- **CTD Grading Scale**

A+	100-97%	A	96-93%	A-	92-90%
B+	89-87%	B	86-83%	B-	82-80%
C+	79-77%	C	76-73%	C-	72-70%
D+	69-67%	D	66-63%	D-	62-60%
F	below 60%				

- **Breakdown of Final Grade**

Tests/Quizzes (50%), Homework/MPs (20%), Final Project (15%), Final Exam (15%)

- **Post-Assessment**

Practice AP Computer Science A exam.

Schedule

Date	Topics	In-class Activities	Assignments/Assessments
Monday	<ol style="list-style-type: none"> 1. Computer Science fundamentals 2. The JAVA Language main structures and libraries 3. Hardware (CPU, memory, peripherals) 4. Software components (OS, compilers, virtual machine) 5. Computing: social and ethical issues 	<ol style="list-style-type: none"> 1. Insides of a computer 2. Ethics discussion 	<ol style="list-style-type: none"> 1. Pretest 2. MP 0 – Introduction to JAVA 3. Worksheet 1: Basics 4. Personal Beliefs on computer ethics
Tuesday	<ol style="list-style-type: none"> 1. JAVA Built-in types (int, double, boolean, char) 2. Basic type casting 3. Input / output 4. Basic Strings 5. Boolean expressions 6. Branches (if, if...else, multi-branch, switch) 	<ol style="list-style-type: none"> 1. Boolean Math Worksheet 2. Control Flow Diagrams 	<ol style="list-style-type: none"> 1. Worksheet 2: Branches 2. MP1A - Branches
Wednesday	<ol style="list-style-type: none"> 1. More if-then statements 2. While Loops 3. For-loops 4. Do - While Loops 5. Control Statements (when to use different control flow structures) 6. Analyzing the running time of loops 	<ol style="list-style-type: none"> 1. Running time group analysis 	<ol style="list-style-type: none"> 1. Worksheet 3: Loops 2. MP1B – Control Flow 3. Branch Quiz
Thursday	<ol style="list-style-type: none"> 1. Writing and calling static methods 2. Formal and Actual Parameters 3. Method Decomposition (top/down design) 4. String Class Methods 5. Math Class 	<ol style="list-style-type: none"> 1. Breaking it Down Activity 2. Investigation of the Math and String Class 	<ol style="list-style-type: none"> 1. Worksheet 4: Methods 2. MP2 – Methods and Control Flow 3. Control Flow Quiz
Friday	<ol style="list-style-type: none"> 1. Recursive Methods (including conceptual introduction to recursive sorting algorithms) 2. Comparing recursion and iteration (in terms of comprehension, speed, resources) 3. Review 	<ol style="list-style-type: none"> 1. Review Game 2. Comparing and Contrasting Recursion and Iteration 	<ol style="list-style-type: none"> 1. Worksheet 5: Recursion 2. Week 1 Review 3. MP3: Recursion and Iteration

Date	Topics	In-class Activities	Assignments/Assessments
Monday	<ol style="list-style-type: none"> 1. Declaring user defined types 2. Instance variables 3. Constructors 4. Instance methods (encapsulation) 5. Access specifications (pub. / private) 6. GridWorld Part 1 	<ol style="list-style-type: none"> 1. GridWorld: Opening Investigation 	<ol style="list-style-type: none"> 1. Worksheet 1: Objects and Classes 2. MP4: Your Own Type 3. GridWorld Part I exercises 1 – 4 4. Week 1 Exam
Tuesday	<ol style="list-style-type: none"> 1. More object-oriented design 2. Designing a class from requirements using pre/post conditions 3. Method overloading 4. Wrapper Classes (Integer, Double) 5. Object relationships 6. Left-over GridWorld Part I discussion 	<ol style="list-style-type: none"> 1. To Please a Client: Object Design Activity 	<ol style="list-style-type: none"> 1. GridWorld Worksheet 2. Worksheet 7: More Objects 3. MP5: More with Objects
Wednesday	<ol style="list-style-type: none"> 1. Arrays/ArrayLists (declaring, initializing, indices, length) 2. Traversals (“for each”), insertions, deletions with arrays and ArrayLists 3. Sorting (insertion, selection, merge) 4. Searching (sequential, binary) 	<ol style="list-style-type: none"> 1. Group Activity on sorting and searching 	<ol style="list-style-type: none"> 1. Worksheet 8: Arrays 2. MP6: Arrays and ArrayLists 3. Objects Quiz
Thursday	<ol style="list-style-type: none"> 1. Comparing running times of search algorithms presentation 2. Basic Inheritance 3. What is inheritance? 4. How to extend a class 5. Class hierarchies 6. Method Overriding 	<ol style="list-style-type: none"> 1. Group Activity on sorting and searching 	<ol style="list-style-type: none"> 1. Worksheet 9: Inheritance 2. MP7: Simple Inheritance 3. Array Quiz
Friday	<ol style="list-style-type: none"> 1. Abstract Classes 2. Polymorphism and casting 3. Interfaces (Comparable, Locatable) 4. Common interfaces 5. When to use abstract classes and when to use interfaces 	<ol style="list-style-type: none"> 1. Exploring GridWorld Part II 2. Review Activity 	<ol style="list-style-type: none"> 1. Worksheet 10: More Inheritance 2. Read GridWorld Part II 3. Inheritance Quiz
Monday	<ol style="list-style-type: none"> 1. Grid World Case Study Part II (extending Bugs) 	<ol style="list-style-type: none"> 1. Circle Bug, Spiral Bug, Z-Bug, and Figure 8 Bug Handout 2. Final Project Coding 	<ol style="list-style-type: none"> 1. GridWorld Part II: Question Set II 2. Decide Final Project 3. Week 2 Exam
Tuesday	<ol style="list-style-type: none"> 1. GridWorld Case Study Part III (Location Interface and interaction with Actors) 	<ol style="list-style-type: none"> 1. Polite Bug and Jumper Bug Handout 2. Final Project Coding 	<ol style="list-style-type: none"> 1. GridWorld Part III Set 3, Set 4, Set 5 2. GridWorld Part II Quiz
Wednesday	<ol style="list-style-type: none"> 1. GridWorld Case Study Part IV Day 1 (Dissecting Chameleon Critter) 	<ol style="list-style-type: none"> 1. Cannibal Critter, RockCritic KingCrab Critter Handout 2. Final Project Coding 	<ol style="list-style-type: none"> 1. GridWorld Part III Quiz 2. Retake Object Oriented Exam (if necessary)
Thursday	<ol style="list-style-type: none"> 1. GridWorld Case Study Part 4 Day 2 (Design your own Critter) 2. Final Exam 	<ol style="list-style-type: none"> 1. GridWorld: Design your own Critter 2. Final Project Coding 	FINAL EXAM
Friday	<ol style="list-style-type: none"> 1. Show-and-Tell with final project 2. Miscellaneous makeup work 3. Ethics Discussion 	<ol style="list-style-type: none"> 1. Following Up on Computing Ethics 2. Exploring the final Projects 	

CTD Statement on Third-Party Web Sites

Instructors are required to thoroughly review any third-party web sites they intend to use in their courses for inappropriate content. However, because web content continuously changes, CTD disclaims any responsibility for any of the content contained on third-party web sites used in course materials. If you become aware of anything that may be inappropriate, please notify CTD staff immediately.

SAMPLE