AP® Computer Science A
2009 Tentative Outline

Please note: this outline is bound to change SIGNIFICANTLY according to student interest as well as to accommodate guest speakers and other unknowns.

Practice-It problems can be found at http://webster.cs.washington.edu:8080/practiceit/

Week	Topics	Essential Questions	Activities
1	Hardware Software Output Compilers Classes	What is Computer Science? What does a basic program look like? How do we display information to a user? What is Programming? What are programming languages? What are computers?	Dissect a PC. What are the criteria for something to be a computer?
2	Algorithms Comments Methods Naming Debugger Error-checking	How do we decompose problems into algorithms? How is program complexity managed? How do we group related tasks? How do we name our programs and identifiers? How do we make sure we can read our programs later?	Compare ugly and nice code Project1: Song
3	Variables RAM Primitive types Casting Binary Expressions Modulus Objects	How do we manage data in our programs? How is data represented in a computer? Why are ints and doubles different types? What kinds of operations can we complete on primitive types? What is the difference between primitive types and objects?	Practice-It: expressions, variables2, many stars Precision error exploration Count in binary Establish coding conventions
4-5	Definite loops Nested loops Constants Parameters Input Cumulative sum Ethics - DoS	How do we make parts of our program repeat? How do we represent patterns that change in two dimensions? How do we give our algorithms flexibility based on user input? When can loops be dangerous?	Practice-It: Fibonacci, number triangle, wave numbers 40 Project2: Space Needle Denial of Service attacks discussion
6	Algorithm design Graphics More Objects Return	How do we draw graphical elements on the screen? How do objects interact with each other? How do we use information from a method later?	Project3: drawings

7-8	If/Else Java classes Using the API Java.lang.Math String char	How do we make our program behave differently based on conditions? How do we perform more complex math operations? How do we manipulate text in Java?	Project4: Calorie tracker Exam 1
9	Indefinite loops Boolean Math.random Probability	How do we make parts of our program repeat a number of times we don’t know ahead of time? How do we model real-life situations with uncertainty? How do we get and use random values?	Multiplication tutor Practice-It: while loop mystery, gcd, random walk Project5: Guessing Game
11-12	OOP Constructors Encapsulation Fields toString Inheritance	How do we model real-world concepts? How do we model relationships between real-world objects that share common traits? How do we break code up into multiple files?	Point class Practice-It: TimeSpan Discussion: Converter class (BJ p 175)
13-14	GridWorld I Writing Classes Ethics: GPL	What do large programs look like? How is source code licensed?	Case Study Part 1 Intellectual Property
15	File input	How do we read in data from external sources?	Practice-It: fixSpacing, Project6: Baby Names
17		Review	Semester exam

1		What topics are we comfortable with? Which do we need more practice on? How do the programming building blocks we have learned help solve CS problems? What do industry professionals do?	Topic summary Sample AP Industry guest
2	Arrays ArrayLists Data Structures	How do we represent a series of related values? How do we retrieve and manipulate values from a list?	Project7: Personality Test
3	Recursion	What are the differences between recursion and iteration?	Project8: Sierpinski
4-5	Gridworld II Interfaces	How can we separate functionality from implementation? How do we capture relationships between classes? How can programs use different types of objects interchangeably?	Project9: Gridworld Tic-Tac-Toe

6-7	Runtime analysis Search Sort Comparable List interface	How do we compare different solutions to the same problem? What are different ways to sort data? What are different ways to search though data?	Project10: iTunes sorting
8-9	Object interaction Debugging	How do we approach designing a multi-class system? How do we make sure that our program is working properly?	Project11: Shopping
10	Gridworld III		
11-15	AP Review	What does the AP test look like? Why would we ever want to know how to code on paper? What deep concepts does the test emphasize? What are test-taking strategies?	Practice tests