

AP COMPUTER SCIENCE A SYLLABUS

Course Resources

Lambert, Ken, and Martin Osborne. Fundamentals of Java, AP Computer Science Essentials for the A and AB Exams. 3rd ed. Boston: Course Technology, 2007.

Henrickson, Poul, Kolling, Michael, Greenfoot, University of Kent, www.Greenfoot.org

Parlente, Nick www.JavaBat.com Interactive learn Java website

College Board. AP GridWorld Case Study. New York: College Entrance Examination Board, 2007.

Barnes, David J., Kolling, Michael J., Objects First with BlueJ

Course Planner [C2]

The resources list includes the following text references: www.Greenfoot.org (Greenfoot), Fundamentals of Java (FJ) and Objects First with BlueJ (OFBJ), GridWorld Case Study (GW).

C2 The course includes all the topics listed in the 'Computer Science A' column of the Topic Outline in the AP Computer Science Course Description



C6 The course teaches students to code fluently in an object oriented paradigm using JAVA. The course teaches students how to use standard JAVA library classes



C3 The course teaches students to design and implement computer-based solutions to problems in a variety of application areas



C5 The course teaches students to develop and select appropriate algorithms and data structures to solve problems

C4 The course teaches students to use and implement commonly used algorithms and data structures

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)	
1 (0-3)	GreenFoot (Introduces objects) Topics [C3] Objects Classes Methods Looping Conditionals Class template Objectives: Write and use simple classes with <u>GreenFoot</u> Wombat, Crab, Space Invaders scenarios Learn the basics of conditionals and looping	Resource: <u>GreenFoot</u> tutorials and <u>GreenFoot</u> <u>GreenFoot</u> introduction video; create new scenario video; making things move video Assessments: Create Wombats, Leaf objects in <u>GreenFoot</u> Use World, Actor, <u>GreenFoot</u> Image class Create a new Rock class Make Wombats act; invoke Wombat methods Add a turn_random method to Wombats Use loops to make Wombats continue, repeat Use conditionals to make decisions Repeat process with Crab, Space Invaders scenarios		Notes: <u>GreenFoot</u> Learning by coding, running, modifying and seeing the results graphically is the essence of <u>GreenFoot</u>. Try to bring what you learn visually in <u>GreenFoot</u> into your programming when using an Integrated Development Environment like <u>BlueJ</u> or <u>JCreator</u> Acquire an intuitive or 'gut' feeling for objects, classes, methods, looping and conditionals using <u>GreenFoot</u> Quiz on Objects, Classes, Methods
2 (4)	Java Basics Topics [C3][C4] Computer basics Java basics Using compiler and IDEs Input and Output String basics Objectives: Understand terminology: CPU, system and application software, LAN, WAN, hard disk, CD-ROM, USB Flash Drives Computer Ethics: paying for intellectual property Understand terminology: compiler, IDE, JVM Edit, compile, and run simple programs in JAVA Understand different compile time errors, runtime errors, and logic errors. Use Scanner class Use output with <u>System.out</u> using print and println. Practice Strings interactively at <u>Javabat</u>	Resource: FJ: Chapters 1-2 FJ Project 1-1 Critical Thinking FJ Chapter 2 Review; FJ pg 162-163 <u>JavaBat</u> Assessments: Labs: <u>Project 2-3 Kilometers to nautical miles</u>, <u>Project 2-5 Momentum</u>. Create a new class <u>ConvertCtoF</u> that inputs a Celsius temp and outputs its Fahrenheit equivalent value, using the <u>Convert</u> class (pg 42) as starting point. Strategies: Scenario: Computer ethics and paying for intellectual property. Class is given the opportunity to decide how best to copyright and to protect a simple computer game scenario they created using <u>GreenFoot</u>. Assign a lot of small programs that illustrate different types of input and output. Practice with Strings at <u>JavaBat</u>		Notes: Read chapters 1 and 2 Programming is different than learning English or History; it is more like learning math. You will learn more by trying to create programs that put into application or use the concepts you are learning; do that as much as you can. Try to absorb the terminology as you are introduced to it and use it, rather than after the fact memorization. Solve at least 2 String examples at <u>JavaBat</u>. Complete the assigned Labs Quiz Chapter 1 Quiz Chapter 2

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)
3 (5-6)	Defining Variables, Arithmetic Expressions Topics [C3][C4][C5][C6] Using and understanding variables Comments <u>Arithmetic expressions in JAVA programs</u> Representing numbers in different bases String expressions and methods GUI panels Objectives: Understand terminology: comments, variables, constants, reserved words, literals <u>Declare and initialize variables and constants in JAVA</u> Understand mathematical expressions in JAVA and operator precedence Use casting to make data more accurate <u>Understand finite numbers; range of integers, real, float</u> Use assignment operator correctly	Resource: FJ: Lesson 3 Assessments: Case Study 1: Income Tax Calculator Projects: 3-1 Area Cube; 3-2 Sphere; 3-3 Kinetic energy; 3-4 Weekly Pay; 3-5 Overtime; Example 3.6 (pg 97) Strategies: Students need practice using different types, double and int in mathematical operations. Present a lot of small program examples in which they have to find errors. Learn basic Graphics and <u>GUIs</u>	Notes: Read chapter 3 Using <u>JCreator</u> , run the Income Tax Calculator <u>Do Exercise 3.2 (1-25) and Exercise 3.3</u> Use the student data files to run Case Study 1; Complete the assigned Labs. Page 92; play with panels. Create some GUI panels by making JAVA Classes Quiz: mathematical expressions; operator precedence, ranges of numeric types and assignment operator Quiz: 80 point quiz (based on exercise 3.2)

4 (7-10)	Conditionals and Looping Topics [C3][C6] if, else, while, for loops Objectives: Understand terminology: control statements, counter, infinite loop, iteration, nested loops, logical operators, truth tables <u>Construct syntactically correct loops and conditional statements</u> Understand the different loop errors and debugging techniques (hand tracing and extra print statements) <u>Use logical operators to make programs more robust and responsive</u> Construct truth tables <u>Be able to calculate statement execution counts, iterations?</u>	Resource: FJ: Chapter 4 <u>GreenFoot Breakout</u> ; <u>Pong</u> ; <u>Wombat</u> ; Crab scenarios <u>JavaBat</u> Assessments: Labs: Projects Triangle 4-2; Telephone Call 4-3; Population growth 4-6 Strategies: Students need practice writing different types of loops. Interactively practice writing loops at <u>JavaBat</u> Make changes to the loops in <u>GreenFoot</u> scenarios Breakout and <u>Pong</u>	Notes: Read all of chapter 4, trying to create short working classes from the examples you discover in the book. Consider the shorter examples as pieces of a larger puzzle, pieces you will need for the Projects. Learn both the uses and terminology of loops Complete the Projects (Labs) Learn to write loops interactively at <u>JavaBat</u> ; practice writing loops; solve at least 2 new problems that use loops. Loops at work. Test Breakout and <u>Pong</u> using <u>GreenFoot</u> . Modify the loops and watch the results! Quiz 80 point quiz on loops.
-------------	---	---	---

AP COMPUTER SCIENCE A SYLLABUS

Unit weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)
5 (11-12)	Introduction to Classes and OOP Topics [C4][C5][C6] Creating and using classes Explore <u>GreenFoot</u> MBS scenarios Objectives: Understand terminology: constructor, accessors, mutator, instance variable, encapsulation, information hiding, procedural abstraction <u>Understanding the difference between public and private access in a class</u> Use and comprehend the <u>DecimalFormat</u> class and the Random class <u>Write classes from scratch, choosing appropriate data representation</u> Understand how to declare a method and declare parameters in that method <u>Understand the use of preconditions, postconditions and assertions when designing methods</u> Understand scope, lifetime	Resource: FJ: Chapter 5 FJ: Chapter 5 Review (pg 195) <u>GreenFoot</u> MBS scenarios; Moon scenario Assessments: Add new classes and methods to MBS simulations in <u>GreenFoot</u> Labs: Case Study 1 Student Test Scores; <u>Proj 5-4</u> Add Dice class to Lucky Sevens; <u>Proj 5-5</u> <u>BankAccount</u> or 5-6 Library Strategies: Give students classes to complete in which they are given a description and they must choose appropriate representation for that class. Utilize the interactivity and graphics capabilities of <u>GreenFoot</u> to solidify understanding of JAVA classes using three scenarios based on the Marine Biology Case Study	Notes: Carefully read chapter 5 with an eye to the terminology surrounding JAVA classes. Do Exercise 5.2 (1-11); Exercise 5.4; Exercise 5.5 Tinker with, modify, and take apart the Marine Biology Case Study scenarios in <u>GreenFoot</u>, learning to identify the parts of a Class and basics of OOP. Case Study 1 as a class Complete the Projects individually Complete chapter 5 Review Quiz Chapter 5 (based on Ch 5 Review)

6 (13)	More on Control Statements Topics [C3][C6] Understanding truth tables Creating and evaluating Boolean expressions Nested and extended if statements Improving User interface Objectives: Construct complex Boolean expressions using logical operators Understand the logic of nested if statements and extended if statements Learn to create menu driven programs	Resource: FJ: Chapter 6 FJ: Chapter 7 <u>JavaBat</u> Assessments: Labs: Case Study 1 Compute Weekly Pay; Case Study 2 Fibonacci Numbers Strategies: Give students the chance to solve larger problems that use nested loops More loops at <u>JavaBat</u>	Notes: Do Exercise 6.1 (examples 1-2 truth tables); Exercise 6.2, test data if statements; Exercise 6.3 (1); Exercise 6.4 nested ifs; Exercise 6.5 output nested ifs; writing nested if code <u>seq.</u>; Exercise 6.7, writing assertions Begin Case Study 1 as a class; complete Case Study 2 individually construct Boolean expressions at <u>JavaBat</u> and work with nested if statements; at least 2 Quiz: Chapter 6 quiz Skim Chapter 7 and try out the examples in the student data files
-----------	--	--	--

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)
7 (14)	Arrays Topics [C4][C5][C6] Declaring and initializing arrays Manipulating arrays with loops Creating parallel arrays Objectives: Understand terminology: array element, index, logical size, physical size, parallel arrays Declare one-dimensional arrays in JAVA Use initializer lists when declaring arrays Manipulate arrays using loops and array indexes Use physical and logical size of an array together to guarantee they do not go beyond the bounds of their array Understand advantages of parallel arrays Learn to use arrays of primitive data types as well as arrays of objects Understand when to use array or <u>ArrayList</u>	Resource: FJ: Chapter 9 <u>JavaBat</u> Assessments: Labs: Case Study Test Scores Using Arrays Proj 9-1 create arrays <u>oddList</u> , <u>evenList</u> or <u>Proj9-2 array averageDoubles</u> ; <u>Proj 9-7 magicSquare array</u> or <u>pennyPinch array</u> <u>Proj 9-10</u> Strategies: Students need practice with loops that work with data in arrays Students need to be able to move around an array using indexes properly	Notes on Chapter 9 Arrays Read Chapter 9 a section at a time; then try: Do Exercise 9.2 Ex. 1 Exercise 9.3 Ex 1-4 Exercise 9.4 Ex 3 From the student data files, run Ex 9.1 and 9.2 on pg 315-316; Complete Exercise 9.7 Ex 3 Exercise 9.9 Ex 1 From student data files, run Case Study Test Scores Using Arrays Do Review Questions page 344 Choose only two of the four Projects and write a working JAVA program. Practice solving array problems at <u>JavaBat</u> ; at least 3 new ones.

8 (15-16)	Arrays Continued Topics [C4][C5][C6] Common String methods Searching and Sorting bubble, Selection, Insertion sort Sequential and Binary searches merge sorts Objectives: Write a method for searching an array <u>Perform insertions and deletions at given positions in arrays</u> Trace through sorting and searching algorithms and understand time constraints of each <u>Understand the algorithms behind each of the following searching and sorting techniques: bubble, selection and insertion sorts; sequential search and binary search</u> Understand the time efficiency of each Learn to choose the best search or sort method for each problem	Resource: FJ:Chapter 11-1 to 11-5 Assessments: Labs: Run most of the code examples and do most of the exercises up through section 5 Strategies: Give students examples with which to run through the examples of key concepts of the chapter. Utilize the interactivity and audio -visual capabilities of <u>GreenFoot</u> to solidify understanding of JAVA classes using Turtle Graphics and Shapes scenarios	Notes: Run code example 11.1 and example 11.2 using strings; check for code to run them in student data files Do Exercise 11.1 and Exercise 11.2; complete Exercise 11.3 Run code example 11.3 selection, bubble, insertion sorts and code example 11.4 insertion and removal of array elements. Complete Exercise 11.4 and Exercise 11.5 Arrays of Objects Try solving a few more array challenges at <u>JavaBat</u> ; be sure they are ones you haven't already tackled.
--------------	---	--	--

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)
9 (17-19)	More on Classes Topics [C4][C5][C6] <u>polymorphism</u> Inheritance Abstract class Class static method Class static variable Interfaces super Objectives: know when to use static variables and methods in a class Understand Java interfaces <u>polymorphism</u> define and use methods that have <u>pre</u> and post conditions; throw exceptions	Resource: FJ: Chapter 10 Lambert Turtle Graphics <u>GreenFoot</u> Turtle Graphics; Shapes; Piano; animation scenarios Assessments: Labs: Strategies: Give students examples with which to run through the examples of key concepts of the chapter. Utilize the interactivity and audio -visual capabilities of <u>GreenFoot</u> to solidify understanding of JAVA classes using Turtle Graphics, Shapes, and Piano scenarios	Notes: Use Lambert Turtle Graphics to explore programming examples using <u>polymorphism</u>, inheritance, abstract classes Read chapter 10, completing the following exercises; Do Exercise 10.1; interfaces – complete Exercise 10.3; write an interface, do Exercise 10.4; Exercise 10.5 on class hierarchies; experiment with animation scenario in <u>GreenFoot</u> to see 'super' at work. Complete Exercise 10.6 and explore abstract classes with Lambert Turtle Graphics. Do Exercise 10.8 on parameters; 10.9 <u>pre</u> and post conditions; 10.10 exceptions; do 10.11 Experiment with Classes using piano and animation scenarios in <u>GreenFoot</u>.

10 (20-22)	<u>ArrayList</u> Topics [C4][C5][C6] Using <u>ArrayList</u> Searching and Sorting Strings Objectives: write methods for searching arrays understand why a sorted array is faster to search write insertion and removal methods Understand arrays of objects Perform simple operations with <u>ArrayList</u> class Learn more about String class	Resource: FJ: Chapter 11-6 to end <u>GreenFoot</u> Wombats scenario <u>JavaBat</u> Assessments: Chapter 11 Review; <u>Proj 11-1</u>; <u>Proj 11-2</u>; <u>Proj 11-5</u>; <u>Proj 11-6</u>; <u>Proj 11-7</u> Strategies: Learn about all the different types of arrays by creating programs that use them. Modify Leaf <u>ArrayList</u> in <u>GreenFoot</u> Wombats	Notes: Do Chapter 11 Case Study: <u>Building a Deck of Cards</u> (check student data files for starting code) Do Chapter 11 Review Do <u>Proj 11-1</u> words in sentence; <u>Proj 11-2</u> array 10 numbers; <u>Proj 11-5</u> Game of War; <u>Proj 11-6</u> Gui for War; <u>Proj 11-7</u> Card Images Solve at a few more new array problems at <u>JavaBat</u>
---------------	--	---	--

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)	
11 (23-24)	Recursion, Complexity, Searching and Sorting Topics [C4][C5][C6] Binary Search Big-O notation Infinite Recursion Merge Sort Quick Sort Objectives: Learn about binary searches and how to determine their efficiency Experiment with different sort methods	Resource: FJ: Chapter 12 JavaBat Assessments: Labs: None Strategies:		Notes : Run code example 12:1 and 12:2 (Check student data files) Complete Exercise 12.1 Recursion and Exercise 12:2 recursive sum methods, O methods; Complete Exercise 12.4 quicksort Run Case Study Comparing Sort Algorithms Run code example 12.4; run code example 12.5; run code example 12.6 Complete at least 3 new recursion problems at JavaBat

12 (25-27)	<u>Gridworld (Part 1)</u> Topics [C2][C3][C4][C5][C6] Introduction to GridWorld Case Study Attributes and behavior of actors Define Bug variations Objectives: Observe and experiment with GridWorld Explore Actor State and Behavior Investigate methods of the Bug Class Extending the Bug Class Runner Classes BoxBugRunner	Resource: Gridworld Case Study, Part 1 and Part 2 Assessments: Strategies:		
----------------------	--	---	--	--

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)	
13 (28 -29)	<u>Gridworld (Part 2)</u> Topics [C2][C3][C4][C5][C6] Classes and Interfaces Interacting Objects Objectives: Learn about Classes and Interfaces using <u>GridWorld</u> Use, understand Location class Understand the Grid Interface Getting to know the Actor class Using the Critter Class Understanding Critter behavior Extending the Critter class	Resource: <u>GridWorld</u> Case Study Parts 3 and 4 Assessments: Strategies:		Notes :

14 (30-32)	Review Topics Topics [C2][C3][C4][C5][C6] Review AP Computer Science A topics Objectives: Continue exam preparation	Resource: Review Previous Free Response Questions; AP Study Practice; AP section <u>JavaBat</u>		Notes : Lambert book includes option in <u>ExamView</u> automated testing software to format quizzes and tests to appear as AP format. Continue to practice and also ask students to complete practice tests without the computer, using paper and pencil.
----------------------	--	--	--	---

AP COMPUTER SCIENCE A SYLLABUS

Unit (weeks)	Title, Topics and Student Objectives	Resources, Assignments and Strategies	Date(s)	
15 After AP Exam	Adventure Game Create New <u>GreenFoot</u> Scenarios	Resource: FJ: Chapter 13 <u>GreenFoot</u> Assessments: Strategies:		Notes :