AP® Computer Science A Sample Syllabus 1

Syllabus 1172785v1

Curricular Requirements		Page(s)
CR1	The course teaches students to design and implement computer-based solutions to problems.	2, 3, 4
CR2a	The course teaches students to use and implement commonly used algorithms.	4
CR2b	The course teaches students to use commonly used data structures.	4
CR3	The course teaches students to select appropriate algorithms and data structures to solve problems.	4, 5
CR4	The course teaches students to code fluently in an object-oriented paradigm using the programming language Java.	3, 4
CR5	The course teaches students to use elements of the standard Java library from the AP Java subset in Appendix A of the AP Computer Science A Course Description.	1, 2, 4
CR6	The course includes a structured lab component comprised of a minimum of 20 hours of hands-on lab experiences.	1
CR7	The course teaches students to recognize the ethical and social implications of computer use.	3

Course Overview

Computer Facilities

Our classroom is also our lab—we find this to be very conducive to learning. We have our computers around the outside of the room, with the center set up in a traditional classroom fashion. Our lab and the labs around campus are managed and maintained by a full-time tech staff. They save us countless hours and ensure that we are up and running 100 percent of the time. This course is on a tight schedule; any downtime during lab is extremely detrimental to student learning, as a minimum of 20 hours of course time is dedicated to hands-on labs. **[CR6]**

CR6— The course includes a structured lab component comprised of a minimum of 20 hours of hands-on lab experiences.

Texts

Bergin, Joseph, et al. *Karel J. Robot: A Gentle Introduction to the Art of Object-Oriented Programming Using Java*. Copyright Joseph Bergin.
http://csis.pace.edu/~bergin/KarelJava2ed/Karel++JavaEdition.html

Litvin, Maria and Litvin, Gary. *Java Methods: Object-Oriented Programming and Data Structures, Second AP Edition*. Skylight Publishing, 2011.
http://www.skylit.com

Course Outline

Unit 1

Introduction to the principle concepts in computer science using Karel J. Robot.

Objectives/Topics To Be Covered

- Computer lab, accounts, and an IDE
- Object-oriented programming and top-down design/refinement of individual tasks
- Basic class structure including instance variables, local variables, parameter passing, scope, public/private visibility, use of super
- Sequence, selection, and iteration
- Recursion
- Inheritance and polymorphism, overriding methods
- `java.lang.Math.random()` **[CR5]**
- Analyze, design, code, and test software
- Error categorization/correction

CR5— The course teaches students to use elements of the standard Java library from the AP Java subset in Appendix A of the AP Computer Science A Course Description.

Teaching Strategies

I teach computer science concepts so that students have immediate visual feedback— at least in the beginning. They will truly understand what they have done right and wrong because they can see it. Students should not lose sight of computer science as they examine the details of the computer language. This undertaking is not too difficult since algorithms that solve a variety of robot tasks are both plentiful and provocative, as are the topics of study associated with them. I place emphasis on having creativity and imagination be their guides. My goal for students is to enjoy computer science at the level that it is most inspiring—the conceptual level.

References/Readings
Karel J. Robot and many other related ideas at the author's site.
http://csis.pace.edu/~bergin/KarelJava2ed/Karel++JavaEdition.html

Java Methods, selected readings from Chapters 2, 7, and 8

Go to the class website for a sample daily schedule, PowerPoint presentations, homework, labs, and review exercises.

Assignments/Labs
- Transcribe, compile, and test a program that uses Newton's method to compute square roots. **[CR1]**
- Go to the class website for the daily schedule, which includes homework assignments, labs, review exercises, PowerPoint presentations, and tests.

CR1— The course teaches students to design and implement computer-based solutions to problems.

Unit 2
Java basics

Objectives/Topics To Be Covered
- Source, bytecode, compilers, interpreters, Java virtual machine, platformindependence
- Computer software and hardware components, operating systems
- Assignment statement, primitive data types
- Arithmetic operators, ArithmeticException, precedence, casting/promotion
- `java.lang.Math (abs, pow, sqrt, random)`, static methods **[CR5]**
- Parameter passing terminology and concepts
- String class, object references, aliasing **[CR5]**
- Selection in more detail
- Object is the superclass of all classes
- Interfaces
- Computer ethics and social implications

CR5— The course teaches students to use elements of the standard Java library from the AP Java subset in Appendix A of the AP Computer Science A Course Description.

Teaching Strategies
Classroom discussions on topics of processors, peripherals, and system software are ongoing throughout the course. Students discuss and identify major components and how they interact. They will become familiar with the operations of the hardware and software available in our school and be able to distinguish between a single-user system and a network. It is expected that all students will adhere to the Acceptable Users' Policy given by our district. I introduce interfaces by providing one for students and having them write a couple of classes that implement the interface. In this manner, I am giving their lab/class its basic structure, providing a lab specification, especially if it contains `Javadoc`. It's also a way to automate testing their labs. I am guaranteeing that the students' classes all have the same method signatures, enabling them to easily test all of their methods.

I engage my students in a number of activities and discussions focused on the ethical and social implications of computer use such as protection of privacy, intellectual property, and public safety. I introduce them to both the ACM and IEEE and their published Codes of Ethics. Dr. Jody Paul has an excellent site listing many resources that we also use to facilitate discussion and activities focused on computer ethics. **[CR7]**

CR7— The course teaches students to recognize the ethical and social implications of computer use.

References/Readings
Java Methods, Chapters 1, 3, 5, 6,7, 10, and 15

Jamtester, JUnit, and unit testing www.jamtester.com

Dr. Jody Paul www.jodypaul.com/SWE/ethics.html

Assignments/Labs
- Students are given a program that draws a sequence of differently colored rectangles and are asked to modify the code so that the result will be a sequence of rectangles that gradually change in color from the color of the first to the color of the last. The algorithm to blend the correct color for each rectangle requires the students to use proportions based on the distance each rectangle is from the first and last rectangles. **[CR1]**
- Magpie Lab (activities 1-5) **[CR1] [CR4]**
- *Java Methods*, selected exercises and labs from chapters 1, 3, 5, 6, 7, 10, and 15
- Polygon lab with unit testing **[CR1]**

CR1— The course teaches students to design and implement computer-based solutions to problems.

CR4— The course teaches students to code fluently in an object-oriented paradigm using the programming language Java.

Unit 3
Elevens Lab

Objectives/Topics To Be Covered
- Intercommunicating objects
- Inheritance
- Interfaces (Comparable) and Abstract classes
- Array basics
- Data structure design and selection

Teaching Strategies
We learn how to evaluate and select algorithms, how to relate classes one to another, and how to debug and test our code.

References/Readings
AP CSA Elevens Lab

Assignments/Labs
Elevens Lab with activities and enrichments **[CR1] [CR4]**

Unit 4
Arrays and `ArrayLists` **[CR5]**

Objectives/Topics To Be Covered
- Declaring, constructing, initializing, and indexing arrays/`ArrayLists` **[CR2b]**
- Storing primitives and objects in arrays/`ArrayLists` **[CR2b]**
- Traversing, inserting, deleting array/`ArrayList` elements **[CR2b]**
- Passing arrays/`ArrayLists` to methods
- Wrapper classes—`Double, Integer` **[CR5]**
- Casting, `ClassCastException`, `ArrayIndexOutOfBoundsException`
- 2-D arrays **[CR2b]**

Teaching Strategies
Practice with arrays and `ArrayLists`.

References/Readings
Java Methods, selected readings from Chapters 12 and 13

Assignments/Labs
- Write a program that measures the frequencies with which each letter of the alphabet occurs in a file. **[CR1]**
- Picture Lab with activities and enrichments **[CR1] [CR4]**
- *Java Methods*, selected exercises and labs in Chapters 12 and 13
- Given a program that draws one equilateral triangle, write a program that draws a Sierpinski gasket (a figure that contains nested triangles). **[CR1]**
- Design a class that models a fraction and arithmetic with fractions. **[CR1]**

Unit 5
Searching and Sorting

Objectives/Topics To Be Covered
- Insertion and selection sorts **[CR2a]**
- Binary searching versus Sequential **[CR2a]**
- Introduction to Big-Oh **[CR3]**
- Recursion
- Mergesort **[CR2a]**
- (optional) `java.util.Arrays and java.util.Collections`

Teaching Strategies
While working with the traditional sorts and searches, I introduce some simple Big-Oh concepts and counting. Big-Oh is not part of the AP CSA Exam, but the counting of statements being executed is a part of the Exam. I have students count comparisons done while sorting and then graph the results. We discover why we prefer to measure efficiency by comparing the number of operations to the size of the data set rather than by looking at the program's execution time. I also use the algorithms that they have studied up to

CR5— The course teaches students to use elements of the standard Java library from the AP Java subset in Appendix A of the AP Computer Science A Course Description.

CR2b— The course teaches students to use commonly used data structures.

CR1— The course teaches students to design and implement computer-based solutions to problems.

CR4— The course teaches students to code fluently in an object-oriented paradigm using the programming language Java.

CR2a— The course teaches students to use and implement commonly used algorithms.

CR3— The course teaches students to select appropriate algorithms and data structures to solve problems.

now (e.g., reading data, common array algorithms) to explore Big-Oh.

This is a good place to work recursion back into the course, since we can explore further how the linear and binary searches can be written both iteratively and recursively.

Students will gain additional practice with arrays as they explore the nontrivial task of merging two sorted lists. In addition, students will once again see a comparison between a recursive and non recursive solution to an algorithm. **[CR3]** Now that the students have had a chance to play with all of the sorts and searches in the AP curriculum, I like to introduce them to two more powerful and fun classes, `java.util.Arrays` and `java.util.Collections`. By this time in the course the students are quite adept at reading an API; this gives them a bit more practice.

CR3— The course teaches students to select appropriate algorithms and data structures to solve problems.

References/Readings
Java Methods, Chapters 4 and 14

Big-Oh handout

The xSortLab Applet http://math.hws.edu/TMCM/java/xSortLab

Assignments/Labs
- Worksheets and sample source code—sorting, searching, recursion,counting iterations, analysis
- *Java Methods*, Chapters 4 and 14 for lab ideas

Unit 6
Review

Objectives
- Ensure students know what is coming on the AP Exam
- Earn a 5 on the AP Exam