AP Computer Science A

Course Resources:

- *Java Software Solutions for AP Computer Science A*, J. Lewis, W. Loftus and C. Cocking, 2nd Edition, 2007, Prentice Hall. (Code JSS)
- AP Gridworld Case Study (Code GSC)
- *Computer Science Unplugged*, T. Bell, I Witten and M. Fellows, http://www.canterbury.ac.nz, 2002.
- Video: *Sorting out Sorting*, Ronald Baecker, University of Toronto, 1981.
- Website: *Sorting Algorithms*, Visual Demonstrations and Sample Code, University of British Columbia, http://www.cs.ubc.ca/~harrison/Java/sorting-demo.html
- Website: http://www.bluej.org
- Website: http://www.eclipse.org
- Website: http://www.eimacs.com, Institute for Mathematics and Computer Science (Excerpts)

Overview:

This course blends daily mini-lectures and hands-on programming assignments to build a functional understanding of computer systems, computational thinking, and object-oriented system design. Students will:

- Design and implement computer-based solutions in a variety of application areas.
- Implement commonly used algorithms and data structures.
- Create and select appropriate algorithms and data structures to solve problems.
- Develop coding fluency in an objected-oriented paradigm using subset A of the standard Java library classes.
- Read and understand the AP Computer Science Case Study, a large program consisting of several classes and interacting objects.
- Identify the major hardware and software components of a computer system, their relationship to one another, and the roles of these components within the system.
- Recognize the ethical and social implications of computer use.

Operationally, the course is conducted in the following manner.

- New concepts are introduced daily in brief mini-lectures and demonstrations.
- Students have the immediate opportunity to exercise new knowledge within a small team (Pilot/Navigator) programming assignment.
- Teacher has daily opportunities to provide each student with timely, personal feedback on design and coding techniques.
- As the course proceeds, programming assignments increase in complexity.
- Larger teams (three or four students) are used for Gridworld Case Study (GSC) exploration. These larger teams make class presentations and defend their findings.
- Students complete AP-style Free Response Questions as individual homework assignments to both demonstrate skills and develop comfort with the AP Exam format.

- All topics from the APCS (Subset A) are covered within lectures and labs.

Unit 1, Computer Systems and Programming Environments

References:

- Chapter 1 and Appendix E, JSS
- The BlueJ Tutorial
- Chapter 1, GSC (Observing and Experimenting with Gridworld)

Objectives:

- Describe the relationship between hardware and software.
- Define various types of software and how they are used.
- Identify basic computer hardware and explain what it does.
- Explain how the hardware components execute programs and manage data.
- Describe how computers are connected together into networks to share information.
- Explain the importance of the Internet and the World Wide Web.
- Introduce the Java Programming Language.
- Describe the steps involved in program compilation and execution.
- Introduce graphics and their representations.

Assignments:

- Install appropriate Java Components and the BlueJ IDE at home
- Team assignment: Selected JSS projects, pages 52-53
- Team assignment: Chapter 1, GSC: Do You Know Set #1 and Exercises

Assessments:

- Multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 47-52
- Team presentation of Gridworld Case Study (GCS) assignment

AP Topic Outline Mapping:

- I.A.1 – Read and understand a problem description, purpose and goals
- III.B.1 – Categorize errors: compile-time, run-time, logic
- III.B.2 – Identify and correct errors
- III.B.3 – Techniques: use a debugger, add extra output statements, hand-trace code
- III.H.1 – Representation of numbers in different bases
- VI – Computing in context

Unit 2, Objects and Primitive Data

References:

- Chapter 2, JSS

Objectives:

- Define the difference between primitive data and objects.
- Declare and use variables.
- Perform mathematical computations.
- Create objects and use them.
- Explore the difference between a Java application and a Java applet.
- (If time permits) Create graphical programs that draw shapes.

Assignments:

- Team assignment: Selected JSS projects, pages 119-121.

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 113, 116-119

AP Topic Outline Mapping:

- I.A.2 – Apply data abstraction and encapsulation
- I.A.5 – Identify reusable components from existing code using classes and class libraries
- II.A.1.a – Object-oriented development
- II.B.1 – Primitive types versus objects
- II.B.2.a – Constant declarations
- II.B.2.b – Variable declarations
- II.B.3 – Console output
- II.C – Java library classes: Integer, Double, String, Math, Random
- III.H.2 – Limitations of finite representations (integer bounds, floating point precision, rounding)
- IV.A – Simple data types (int, Boolean, double)
- IV.B – Classes

Unit 3, Program Statements

References:

- Chapter 3, JSS

Objectives:

- Discuss basic program development steps.
- Define the flow of control through a program.
- Learn to use if statements.
- Define expressions that let us make complex decisions.
- Learn to use while and for statements.
- (If time permits) Use conditionals and loops to draw graphics.

Assignments:

- Team assignment: Selected JSS projects, pages 187-189

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 180 and 184-187

AP Topic Outline Mapping:

- I.A.1 – Read and understand a problem description, purpose and goals
- I.B.3 – Choose appropriate data representation and algorithms
- II.A.1.b – Top-down development
- II.B.4.b – Sequential control structure
- II.B.4.c – Conditional control structures
- II.B.4.d – Iterative control structures
- III.H.2 – Limitations of finite representations (integer bounds, floating point precision, rounding)

Unit 4, Writing Classes

References:

- Chapter 4, JSS
- Chapter 2, GCS (Bug Variations)

Objectives:

- Define classes that act like blueprints for new objects, made of variables and methods.
- Explain encapsulation and Java modifiers.
- Explore the details of method declarations.
- Review method invocation and parameter passing.
- Explain and use method overloading.
- Learn to divide complicated methods into simpler, supporting methods.
- Describe relationships between objects.
- (If time permits) Create graphics based objects.

Assignments:

- Team assignment: Selected JSS projects, pages 251-252
- Team assignment: Chapter 2, GCS: Do You Know Set #2 and Exercises

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 244 and 248-250
- AP-style Free Response question, JSS 255-256
- Team presentation of Gridworld Case Study (GCS) assignment

AP Topic Outline Mapping:

- I.A.3 – Read and understand class specifications and relationships (“is-a”, “has-a”)
- I.A.2 – Apply data abstraction and encapsulation
- I.B.1 – Design and implement a class
- I.B.4 – Apply functional decomposition
- II.A.1.c – Implement encapsulation and information hiding
- II.A.1.d – Implement procedural abstraction
- II.B.2.c – Class declarations
- II.B.2.e – Method declarations
- II.B.2.f – Parameter declarations
- II.B.4.a – Using methods as control structures
- III.F.1 – Understand method pre- and post-conditions
- III.F.2 – Assertions
- IV.B – Classes as data structures

Unit 5, Enhancing Classes

References:

- Chapter 5, JSS
- Chapter 3, GCS (Gridworld Classes and Interfaces)

Objectives:

- Define reference aliases.
- Explore passing object references as parameters.

- Learn to use the static modifier.
- Define formal interfaces and their class implementations.
- Define nested classes and inner classes
- (If time permits) Learn about basic graphical user interfaces.

Assignments:

- Team assignment: Selected JSS projects, pages 310-311
- Team assignment: Chapter 3, GCS: Do You Know Sets #3 and #4

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 305 and 309
- AP-style Free Response question, JSS 314
- Team presentation of Gridworld Case Study (GCS) assignment

AP Topic Outline Mapping:

- I.B.1 – Design and implement a class
- I.B.2 – Design an interface
- I.B.3 – Choose appropriate data representation and algorithms
- II.B.2.d – Interface declarations
- II.C – The Comparable interface
- III.A.1 – Test classes and libraries in isolation
- III.A.2 – Identify boundary cases and generate appropriate test data
- III.A.3 – Perform integration testing
- III.B.2 – Identify and correct errors
- III.B.3 – Techniques: Use a debugger, add extra output statements, hand-trace code
- III.E.1 – Understand run-time exceptions

Unit 6, Arrays

References:

- Chapter 6, JSS
- Continue Chapter 3, GCS

Objectives:

- Define and Use Arrays
- Describe how arrays and array elements are passed as parameters.
- Explore how arrays and other objects can be combined to manage complex information.
- Explore searching and sorting with arrays.
- (If time permits) Learn to use multidimensional arrays.
- Examine the ArrayList class.

Assignments:

- Team assignment: Selected JSS projects, pages 383-385
- Team assignment: Chapter 3, GCS: Do You Know Sets #5 and #6

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 377 and 381-382
- AP-style Free Response question, JSS 388-389
- Team presentation of Gridworld Case Study (GCS) assignment

AP Topic Outline Mapping:

- II.C – The ArrayList Class
- III.G.1 – Informal comparisons of running times
- III.G.2 – Exact calculation of statement execution counts
- IV.C – One-dimensional arrays
- V.A.1 – Traversals
- V.A.2 – Insertions
- V.A.3 – Deletions
- V.B.1 – Sequential search operations
- V.B.2 – Binary search operations
- V.C.1 – Selection sort operations
- V.C.2 – Insertion sort operations

Unit 7, Inheritance

References:

- Chapter 7, JSS
- Continue Chapter 3, GCS

Objectives:

- Derive new classes from existing ones.
- Explain how inheritance supports software reuse.
- Add and modify methods in child classes.
- Discuss how to design class hierarchies.
- Define polymorphism and how it can be done.
- (If time permits) Discuss the use of inheritance in Java GUI framework.
- (If time permits) Examine and use the GUI component class hierarchy.

Assignments:

- Team assignment: Selected JSS projects, pages 457-458
- Team assignment: Chapter 3, GCS: Group Activity

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 452 and 456
- AP-style Free Response question, JSS 461-463
- Team presentation of Gridworld Case Study (GCS) assignment

AP Topic Outline Mapping:

- I.A.2 – Apply data abstraction and encapsulation
- I.A.3 – Read and understand class specifications and relationships among the classes
- I.A.4 – Understand and implement a given class hierarchy
- I.A.5 – Identify reusable components from existing code using classes and class libraries
- I.B.2 – Design an interface
- I.B.5 – Extend a given class using inheritance
- II.C – The Object Class
- III.D – Extend existing code using inheritance

Unit 8, Recursion

References:

- Chapter 8, JSS

Objectives:

- Explain the underlying ideas of recursion.
- Examine recursive methods and processing steps.
- Define infinite recursion and discuss ways to avoid it.
- Explain when recursion should and should not be used.
- Demonstrate the use of recursion to solve problems.
- Examine the use of recursion in sorting.

Assignments:

- Team assignment: Selected JSS projects, pages 505-507

Assessments:

- AP-style multiple-choice exam from textbook Test Bank
- Short essay exam addressing AP topics for this unit, JSS 500-501, 504-505
- AP-style Free Response question, JSS 510-511

AP Topic Outline Mapping:

- II.B.4.e – Programming constructs (recursion)

Unit 9, Case Study Completion and Exam Preparation Emphasis

References:

- Chapter 4, GCS (Interacting Objects)
- eIMACS website materials (Excerpts)
- Computer Science A Exam free response questions from previous years
- Computer Science A Exam multiple-choice questions from previous years
- Various AP Computer Science Exam study guides

Objectives:

- Read and understand the AP Computer Science Case Study, a large program consisting of several classes and interacting objects.
- Review concepts from throughout the course in preparation for the AP CS (A) exam.

Assignments:

- Team assignment: Chapter 4 requirements, GCS: Do You Know Sets #7 through #9, Exercises, and Group Activity

Assessments:

- Team presentation of Gridworld Case Study (GCS) assignment
- AP Exam
  or
- Comprehensive Final Exam for those who do not take the AP Exam