



School District of Marshfield Course Syllabus

Course Name: AP Computer Science A PS
Length of Course: Year
Credit: 1 Credit

Program Goal:

Empower learners to be college and career ready through standards-based experiences in the classroom and career-based learning experiences with business and industry partners. Design and implement educational experiences for creating a skilled, knowledgeable, and productive workforce. Learners will engage in competencies that enable them to stay up-to-date with evolving skills as they pursue careers directly out of high school, as technical school degree earners, or as university graduates. Our goal is to develop critical thinkers and collaborative problem solvers, providing connections to the issues and challenges facing our local, regional, and global economies.

Course Description:

Learn college-level programming using Java programming language. Concepts included are structured Java programming style, assignment and logical operators, decision-making, looping, functions, and arrays. Also learn the concepts of data structures, classes, inheritance, recursion, and other advanced topics using an object-oriented approach. Students may seek college credit by taking the Advanced Placement Examination in May.

Text and Resource Legend

The following legend codes are used throughout the course planner to identify textbook, reading reference resources, and various assessment resources contained in the weekly schedule. Big Java 2nd Edition is the primary textbook and the central guide and focus of our learning. In addition to those resources referenced throughout the course planner, I frequently complement the students' day-to-day learning with resources of my own which I continually improve upon, based on my own experiences and the unique needs of the learner.

Primary Text	BJ = Big Java 2nd Edition
Assessments / Strategies	175 = 175 Multiple-Choice Questions in Java LAB = Big Java 2 nd Edition Lab Manual FR* = free response problem solving M/C* = multiple choice problem solving GW = GridWorld Case Study *instructor created free response and multiple choice problem solving exercises, familiarizing students with the AP exam format
Secondary Text	HF = Head First Java, 2nd Edition JM = Java Methods A&AB
[C2]	The course includes all of the topics listed in the “Computer Science A” column of the Topic Outline in the AP Computer Science Course Description.
[C3]	The course teaches students to design and implement computer-based solutions to problems in a variety of application areas.
[C4]	The course teaches students to use and implement commonly used algorithms and data structures.
[C5]	The course teaches students to develop and select appropriate algorithms and data structures to solve problems.
[C6]	The course teaches students to code fluently in an object-oriented paradigm using the programming language Java. The course teaches students to use standard Java library classes from the AP Java subset delineated in Appendices A and B of the <i>AP Computer Science Course Description</i> . (Note: Students who study a language other than Java in AP Computer Science must also be taught to use Java, as specified in the AP Java subset.)
[C7]	The course teaches students to read and understand a large program consisting of several classes and interacting objects, and enables students to read and understand the current <i>AP Computer Science Case Study</i> posted on AP Central®
[C8]	The course teaches students to 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.

[C9]	The course teaches students to recognize the ethical and social implications of computer use.
------	---

Course Planner

Week	Topic	Programming Projects	Assessments/ Strategies
1&2	<ul style="list-style-type: none"> • What is Computer Science? • Problem Solving/Algorithms • Hardware/Software/Networks • Usage Policy Review • Privacy / Intellectual Property • Java basics 	HelloWorld w/ variations javac compiler Console versus IDE BJ: P1.1, P1.2, P1.3 Readings: BJ CH01-02	AP pre-test M/C Quiz [C3] [C4] [C5] [C6][C8] [C9]
3&4	<ul style="list-style-type: none"> • Java Code Structure • Objects, Classes & Methods • Types and Variables • Using the API • Errors • Compile, runtime, logic • GridWorld Case Study CH01 • Run / analyze GridWorld GUI • Responsible use of computer systems • Documentation and Style 101 	BJ: P1.6, P1.8 BJ: P2.2, 2.4 Readings: BJ CH02, HF CH01, JM CH03, CH05, GW Narrative CH01	M/C Quiz FR Quiz LAB(1)(2) 175(2005)2 javadoc practice GW Questions Set 1 GW Exercises Set 1 [C3] [C4] [C5][C6][C8][C9]
5&6	<ul style="list-style-type: none"> • Designing Classes • Constructors / Methods • Variables: Instance fields, Local variables, Parameter variables • Testing Classes • Declarations • Intro: Testing & Debugging • Adding output statements • Handtracing code 	BJ: P3.2, P3.4, P3.6 Readings: BJ CH03, CH10, HF CH02	M/C Quiz FR Quiz LAB(3) 175(2004)11 175(2005)16 [C3] [C4] [C5]
7	<ul style="list-style-type: none"> • Data Types • Primitive vs. Objects • Precision and Casting • Constants • Arithmetic Operators / Precedence • Math class: Static methods • Strings 	BJ: P4.2, P4.4, P4.8 Readings: BJ CH04, HF CH03, JM CH06, CH10	M/C Quiz LAB(4) 175(2004)1,5,8 175(2005)1,3 [C3] [C4] [C5][C6]
8	<ul style="list-style-type: none"> • Conditionals • if, if-else, • Relational Operators • Boolean Expressions 	BJ: P6.2, P6.4, P6.8	FR Quiz LAB(6)

	<ul style="list-style-type: none"> • Predicate methods • De Morgan's Law 	Readings: BJ CH06, JM CH07	[C3] [C4] [C5]
9	<ul style="list-style-type: none"> • Iteration • While and for loops • Scope • Nested Loops • Sentinel values • Fibonacci Sequence 	BJ: P7.4, P7.5, P7.10 Readings: BJ CH07	M/C Quiz LAB(7) 175(2004)12 175(2005)14,15 [C3] [C5] [C6]
10&11	<ul style="list-style-type: none"> • Arrays and ArrayLists • Wrapper classes • Enhanced for Loop 	BJ: P8.1, P8.2, P8.4 Readings: BJ CH08, HF CH03, CH10	M/C Quiz FR Quiz LAB(8) 175(2004)7 175(2005)6,9,15 [C4] [C5] [C6]
12,13,14	<ul style="list-style-type: none"> • Searching and Sorting • Sequential searches • Binary searches • Selection sort • Merge sort • Insertion sort 	BJ: P19.2, P19.4, P19.6, Readings: BJ CH19	M/C Quiz LAB(19) [C3] [C5] [C6]
15,16,17	<ul style="list-style-type: none"> • Inheritance • Super / Sub classes • "isa", "hasa" relationships • Instance fields / Methods • Polymorphism • Overriding methods • Abstract methods • The Object class • GridWorld Case Study Chapter 2 	BJ: P13.2, P13.4, P13.6, P13.8 Readings: BJ CH13, HF CH07, GW Narrative CH02	M/C Quiz FR Quiz LAB(13) 175(2004)14,15 175(2005)23 GW Questions Set 2 GW Exercises Set Pg 12-13: 1,2,3 [C3] [C4] [C5][C6][C7]
18&19	<ul style="list-style-type: none"> • Recursion • Recursive computation • Permutations • Recursive methods • Helper methods 	BJ: P18.2, P18.4, P18.7 Readings: BJ CH18,	LAB(18) 175(2004)6,16 175(2005)7 [C4] [C5] [C6]
20	<ul style="list-style-type: none"> • Exception Handling • Throwing exceptions • Catching exceptions 	BJ: P15.2, P15.4, P15.5 Readings: BJ CH15, HF CH11	M/C Quiz LAB(15) [C3] [C4] [C6]
21&22	<ul style="list-style-type: none"> • Interfaces • Interface types • Class coupling • Class/Interface conversions • Early/Late binding 	BJ: P11.2, P11.3, P11.10 Readings: BJ CH11, HF CH08, JM CH11	FR Quiz LAB(11) 175(2004)2 175(2005)10,11 [C3] [C5] [C6]
23	<ul style="list-style-type: none"> • GridWorld Case Study Chapter 2 • GridWorld Case Study Chapter 3 		GW Questions Set 3 p18 Set 4 p20 Set 5 p22 Set 6 p24 GW Exercises Set

		Readings: GW Narrative CH02, GW Narrative CH03	Pg 13-14: 4, 5 [C3] [C5] [C6][C7]
24&25	<ul style="list-style-type: none"> • Testing and Debugging • Unit testing • Boundary cases • Creating test data • IDE testing and debugging • Black-box testing 	BJ: P10.2, P10.4 Utilize student-generated code for debugging and testing Readings: BJ CH10, HF CH05	LAB(10) [C3] [C4] [C6]
26&27	<ul style="list-style-type: none"> • Designing Classes • Cohesion and coupling • Accessors and mutators • Immutable classes • Side effects • Pre and PostConditions • Static methods • Static fields 	BJ: P9.2, P9.4, P9.6, P9.16(optional) Readings: BJ CH09, HF CH09	M/C Quiz FR Quiz LAB(9) [C3] [C4] [C6]
28	<ul style="list-style-type: none"> • GridWorld Case Study Chapter 4 	Readings: GW Narrative CH04	GW Questions Set 7 p28 Set 8 p30 Set 9 p31 GW Exercises Set Pg 32: 1-6 [C3] [C5][C6][C7]
29,30,31,32	<ul style="list-style-type: none"> • Final Exam Preparation / Review 	AP GridWorld Case Study Supplemental Practice Exams 2005-2006 AP Exams Timed Mock Exam Trials	M/C Quizzes FR Quizzes [C2]
33,34,35,36	<ul style="list-style-type: none"> • Enrichment activities • Course content • Emerging technologies • Current Events • Responsible use of Computer Systems 	Readings: TBD	AP Students Create units for Intro Programming students <ul style="list-style-type: none"> • Lego RCX/NXT Robots • Alice language • Wiki • RSS/podcast portal [C9]

Correlation to AP Topic Outline

I. Object-Oriented Program Design

The overall goal for designing a piece of software (a computer program) is to correctly solve the given problem. At the same time, this goal should encompass specifying and designing a program that is understandable, can be adapted to changing circumstances, and has the potential to be reused in whole or in part. The design process needs to be based on a thorough understanding of the problem to be solved.

Planner Week	Course
A. Program Design	
1. Read and understand a problem description, purpose, and goals.	1,2,3,4
2. Apply data abstraction and encapsulation.	3,4,5,6, 15,16,17 21,22
3. Read and understand class specifications and relationships among the classes (“is-a,” “has-a” relationships).	5,6,15,16,17 23,26,27,28
4. Understand and implement a given class hierarchy.	15,16,17 21,22 26,27,28
5. Identify reusable components from existing code using classes and class libraries.	3,4,15,16,17 21,22
B. Class Design	
1. Design and implement a class.	5,6,26,27
2. Choose appropriate data representation and algorithms.	7,8,9
3. Apply functional decomposition.	15,16,17
4. Extend a given class using inheritance.	15,16,17 23,28
II. Program Implementation	
The overall goals of program implementation parallel those of program design. Classes that fill common needs should be built so that they can be reused easily in other programs. Object-oriented design is an important part of program implementation.	
A. Implementation techniques	
1. Methodology	
a. Object-oriented development	1,2,3,4,7
b. Top-down development	8,9
c. Encapsulation and information hiding	15,16,17,21,22 23,28
d. Procedural abstraction	15,16,17,26,27

B. Programming constructs	
1. Primitive types vs. objects	7
2. Declaration	
a. Constant declarations	5,6
b. Variable declarations	5,6
c. Class declarations	5,6
d. Interface declarations	15,16,17,21,22 23,28
e. Method declarations	5,6,26,27
f. Parameter declarations	5,6,26,27
3. Console output (System.out.print/println)	3,4,5,6
4. Control	
a. Methods	3,4,5,6,26,27
b. Sequential	8,9
c. Conditional	8,9
d. Iteration	8,9
e. Recursion	18,19
C. Java library classes (included in the A-level (AP Java Subset)	3,4,26,27 29-32
III. Program Analysis	
The analysis of programs includes examining and testing programs to determine whether they correctly meet their specifications. It also includes the analysis of programs or algorithms in order to understand their time and space requirements when applied to different data sets.	
A. Testing	
1. Test classes and libraries in isolation.	5,6
2. Identify boundary cases and generate appropriate test data.	5,6,24,25
3. Perform integration testing.	5,6,24,25
B. Debugging	
1. Categorize errors: compile-time, run-time, logic	1,2,3,4,20
2. Identify and correct errors.	1,2,3,4,10,11 12,13,14,20
3. Employ techniques such as using a debugger, adding extra output statements, or hand-tracing code.	5,6,10,11 20,24,25
C. Understand and modify existing code	1-36
D. Extend existing code using inheritance	15,16,17
E. Understand error handling	
1. Understand runtime exceptions	5,6,20

F. Reason about programs	
1. Pre- and post-conditions	20,26,27
2. Assertions	20,24,25,26,27
G. Analysis of algorithms	
1. Informal comparisons of running times	10,11,12,13,14 18,19
2. Exact calculation of statement execution counts	9,10,11,12,13 14,18,19
H. Numerical representations and limits	
1. Representations of numbers in different bases	1,2,3,4
2. Limitations of finite representations (e.g., integer bounds, imprecision of floating-point representations, and round-off error)	7,8,9
IV. Standard Data Structures	
Data structures are used to represent information within a program. Abstraction is an important theme in the development and application of data structures.	
A. Simple data types (int, boolean, double)	7
B. Classes	3,4,5,6,26,27
C. One-dimensional arrays	10,11
V. Standard Algorithms	
Standard algorithms serve as examples of good solutions to standard problems. Many are intertwined with standard data structures. These algorithms provide examples for analysis of program efficiency.	
A. Operations on A-level data structures previously listed	
1. Traversals	10,11
2. Insertions	10,11
3. Deletions	10,11
B. Searching	
1. Sequential	12,13,14
2. Binary	12,13,14
C. Sorting	
1. Selection	12,13,14
2. Insertion	12,13,14
3. Mergesort	12,13,14
VI. Computing in Context	
A working knowledge of the major hardware and software components of computer systems is necessary for the study of computer science, as is the awareness of the ethical and social implications of computing systems. These topics need not be covered in detail but should be considered throughout the course.	

A. Major hardware components	
1. Primary and secondary memory	1,2
2. Processors	1,2
3. Peripherals	1,2
B. System software	
1. Language translators/compiler	1,2
2. Virtual machines	1,2
3. Operating systems	1,2
C. Types of systems	
1. Single-user systems	1,2
2. Networks	1,2
D. Responsible use of computer systems	
1. System reliability	1,2,3,4
2. Privacy	1,2,3,4,33-36
3. Legal issues and intellectual property	1,2,3,4,33-36
4. Social and ethical ramifications of computer use	1,2,3,4,33-36

Primary Resource(s):

Java Software Solutions 9th Edition
 Pearson
 ISBN: 978-0-13-446202-8
 ©2018