High School Course Description for AP Computer Science A

Course Title: AP Computer Science A

Course Number: MTH660

Grade Level: 11-12

Meets a UC a-g Requirement: g-College Prep Elective – Mathematics (pending approval)

Meets High School Graduation Requirement for: 10 Elective Credits – Mathematics, Computer literacy, 5 credits computer literacy

Curricular Area: Mathematics Elective

Length: One year

Prerequisites: Algebra II with a C or better grade

Meets NCAA Requirement: no

Course Description

This course provides students with an introduction to computer science and is designed to prepare the student to pass the AP Computer Science A test. This course includes object oriented programming methodology with an emphasis on problem solving and algorithm development and is meant to be the equivalent of a first semester course in computer science. It also includes the study of data structures and abstraction. Students who complete this class will be more competitive for college acceptance, and have the basic programming skills to be successful in their engineering and mathematics classes.

Alignment

This course is aligned to the California Common Core State Standards for Mathematics. This course fulfills the following objectives. (1) enables students to express algorithms in a standard language (Java); (2) requires students to complete substantial programming projects; and (3) involves the study and mastery of various aspects of computer science.

Instructional Materials

Required Textbook(s)

1. Java software solutions for AP Computer Science A, 3rd edition, Addison-Wesley

Supplemental Materials

1. AP GridWorld Case Study, College Board

   AP* Test Prep Series, Pearson Education

Web Sites

1. CollegeBoard.org

Software

1. .Java Development Kit, Oracle. Web Browser

Exit Criteria

Activities	Percentage
Classwork/participation	20%
Assessments	60%
(may include quizzes, tests, performance tasks, etc.)	
Final Examination	20%
Total:	100%

Development Team

This Course of Study was developed in 2013/14 by Raymonn Brown and Richard Montgomery

Pacing Guide for AP Computer Science A

SEMESTER ONE

Key Assignments:

- Each student will select two projects they will work on individually for semester 2 presentations

Assessments:

- Java library AP* tested Classes, computer science vocabulary and Java code syntax

First Quarter		
Weeks	**Standards/Objectives**	**Concepts/Content**
1-3	Object-Oriented Program Design	Read and understand a problem description, purpose, and goals. Apply data abstraction and encapsulation. Understand class specifications and relationships. Identify reusable components from existing code.
4-6	Program Implementation	Understand and use Inheritance, Polymorphism, and Methodology.
7-9	Programming Constructs	Primitive types vs Objects, Declarations

Second Quarter		
Weeks	**Standards/Objectives**	**Concepts/Content**
1-3	Java library	Classes included in the AP* test
4-6	Program Analysis	Code, compile, and run programs
7-9	Data Structures	Types, Classes, Lists, Arrays

SEMESTER TWO

Key Assignments:

- Two *Power Point* presentations (one in February and one in June). Presentation will detail the design, coding, and implementation of a large scale program that the student has selected and has been approved by the instructor. Students started working on their during semester 1.

Assessments:

- GridWorld Case study, AP* practice tests

Third Quarter		
Weeks	**Standards/Objectives**	**Concepts/Content**
1-3	Student project presentations GridWorld role play	Detail the design, coding, and implementation of student selected project GridWorld code walk thru
4-6	GridWorld case study	Create Populate a world Design Your Own Critter

		Design A "Living" World
7-9	AP* test prep	GridWorld practice

Fourth Quarter		
Weeks	**Standards/Objectives**	**Concepts/Content**
1-3	AP* test prep	3 practice tests
4-6	Final preparation for AP* exam	4th practice test
7-9	Computing in context Student project presentations	Privacy, legal, social, and ethical issues Present the design, coding, and implementation of student selected project

Learning Experiences and Instruction:
Teachers utilize the Direct Interactive Instruction model to introduce new skills and concepts that are essential to the grade level content standards, then reinforce and develop those skills each quarter with the goal of bringing students to mastery by the end of the fourth quarter. All instruction will be based on the "I do, We do, You do" scaffolding model with an emphasis on individual differentiation as needed. Teachers will use a variety of the following:

- Inquiry-based learning
- Engaged reading opportunities
- Think-pair-share
- Reciprocal teaching
- Cloze reading & writing
- Guided reading & writing
- Cognitive modeling
- Questioning strategies
- Graphic organizers/concept attainment
- Student-led groups
- Peer pairing
- Metacognitive learning: self-regulation, goal-setting, self-monitoring, and self-questioning

Support for English Language Learners:
Extra time or modified versions of assignments will be given. The District will provide a language assistant. Additional strategies will be developed through the Response to Intervention plans –such as:

- SDAIE strategies
- Texts/materials in first language.
- Flexible grouping
- Structured engagement
- Peer pairing
- Academic vocabulary development
- Realia

Support for Special Education Students:
Extra time or modified versions of assignments will be given. The District will provide an instructional assistant. Additional strategies will be developed through the Individual Education Plan process – such as:

- Realia
- Texts/materials in first language
- SDAIE strategies
- Flexible grouping
- Peer pairing
- Audio & visual aids
- Individualized academic instruction
- Modified assignments
- Modified texts
- Testing accommodations
- Tutoring (peer & teacher)

Stretching the Lesson for GATE Students:
Differentiated curriculum will be provided to challenge the student and provide the student with opportunities to develop their identified talent. Teachers will use a variety of the following:

- Independent study supplemented with mentoring/tutoring
- Compacting
- Acceleration
- Depth & Complexity icons
- Modified texts
- Modified assignments
- Flexible grouping
- Inquiry-based Learning
- Enriched materials and learning experiences

<end>