COURSE DESCRIPTION
POINT ARENA HIGH SCHOOL

I. DESCRIPTION

COURSE TITLE AND LEVEL:	**AP COMPUTER SCIENCE PRINCIPLES, SECONDARY**
LENGTH OF COURSE:	**ONE YEAR**
AVAILABLE TO STUDENTS AT GRADE:	**SOPHOMORE TO SENIOR**
PREREQUISITES:	**ALGEBRA I**
REQUIRED OR ELECTIVE:	**ELECTIVE**

II. GOALS AND OBJECTIVES

AP Computer Science Principles is designed to be equivalent to a first-semester introductory college computing course. In this course, students will develop computational thinking skills vital for success across all disciplines, such as using computational tools to analyze and study data and working with large data sets to analyze, visualize, and draw conclusions from trends. This course's main focus is to foster student creativity. Students are encouraged to apply creative processes when developing computational artifacts and to think creatively while using computer software and other technology to explore questions that interest them. They will also develop effective communication and collaboration skills, working individually and collaboratively to solve problems, and discussing and writing about the importance of these problems and the impacts to their community, society, and the world.

The course's objectives for students are the following:

- Identify impacts of computing,
- Describe connections between people and computing,
- Explain connections between computing concepts,
- Create a computational artifact with a practical, personal, or social intent,
- Select appropriate techniques to develop a computational artifact,
- Use appropriate algorithmic and information management principles,
- Explain how data, information, or knowledge is represented for computational use,
- Explain how abstractions are used in computation or modeling,
- Identify abstractions,
- Describe modeling in a computational context,
- Evaluate a proposed solution to a problem,
- Locate and correct errors,
- Explain how an artifact functions,
- Justify appropriateness and correctness of a solution, model, or artifact,
- Explain the meaning of a result in context,
- Describe computation with accurate and precise language, notations, or visualizations,
- Summarize the purpose of a computational artifact,

- Collaborate with other students in solving a computational problem,
- Collaborate with other students in producing an artifact,
- Share the workload by providing individual contributions to an overall collaborative effort,
- Foster a constructive, collaborative climate by resolving conflicts and facilitating the contributions of a partner or team member,
- Exchange knowledge and feedback with a partner or team member,
- Review and revise the team's work as needed to create a high-quality artifact, and
- Prepare for the AP exam in computer science principles.

III. METHODS OF INSTRUCTION

A. Lecture, discussion, and demonstrations
B. Role-playing
C. Reading assignments
D. Journal writing
E. Student sharing
F. Compositions and/or reports
G. Use of audiovisual materials
H. Homework assignments
I. Small group instruction

IV. METHODS OF EVALUATION

A. Student participation
B. Written exams and quizzes
C. Student journals
D. Written assignments
E. Student self evaluation
F. Peer evaluations
G. Student projects and assignments

V. TEXTBOOKS

Abelson, Hal, Ken Ledeen, and Harry Lewis. *Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion*, published by Addison-Wesley Professional, Boston, 2008. Available for free download at Bitsbook.com.

Barksdale, Karl and E. Shane Tuner. HTML and JavaScript Basics. 4th ed. published by Cengage Learning, Boston, 2011.

VI. SUPPLEMENTAL TEXTBOOKS AND RESOURCES

A. Various resources from Code.org
B. Scratch programming language from MIT Media Lab at scratch.mit.edu and other resources for Scratch
C. AP Computer Science Principles Course home webpage, at https://advancesinap.collegeboard.org/stem/computer-science-principles
D. Snap! programming language from the University of California, Berkeley at snap.berkeley.edu
E. Lightbot program, at https://lightbot.com/hour-of-code-2015-flash.html
F. Flowcharting tools from Gliffy at gliffy.com
G. Various resources at CS10K Community at cs10kcommunity.org
H. Various resources at Cooperative Learning Resources at cooperativelearningresources.weebly.com
I. Datasets from Data.gov
J. Pencil Code at pencilcode.net
K. *Scratch Curriculum Guide*, by ScratchED, available at http://scratched.gse.harvard.edu/guide/download.html
L. *Exploring Computer Science Curriculum* at exploringcs.org

VI. MATERIALS

A. One computer per student with Internet access.
B. One LCD projector per class
C. Poster paper for student projects
D. Google for Education account

COURSE OUTLINE

I. Impact of Computing (Time: about 3 days)

A. Topics
   1. Understanding of how commonly used technology tools, such as smart phone, GPS, etc., affect students' personal lives and society.
B. Resources
   1. Articles from various websites, such as, technews.acm.org, sciencedaily.com, and nytimes.com
   2. "AP Computer Science Principles Creativity and Global Impact Curriculum Module," College Board, October 2014

C. Activities
   1. Students will write a one-to-two page research paper on the impact of a technology tool on them and/or society
   2. Students will do a quick one-minute summary of their findings to the class using presentation software

D. Readings
   1. Varies depending on the technology tool the student has chosen

II. Algorithms
(Time: about 3 weeks)

A. Topics
   1. Understanding the use of algorithms in problem solving
   2. Development of a step-by-step process for problem solving
   3. How to construct algorithms using pseudocode and flowcharts
   4. How algorithms are used to develop computer programs

B. Resources
   1. Programming Puzzles from Lightbot.com
   2. Flowcharting tool at gliffy.com
   3. Pseudocode reference sheet from the College Board
   4. Various resources at Code.org

C. Activities
   1. Solve puzzles using the Lightbot program
   2. Develop and share puzzles students develop using the Lightbot program
   3. Using pseudocode, students will develop a flowchart for a common daily activity, like waking up in the morning, going to school, etc.; sequencing, iteration, and decision-making will be used
   4. Video on developing simple code for a computer program
   5. Students will write step-by-step instructions to create an origami object using pseudocode that a computer might be able to carry out

D. Readings
   1. Pseudocode reference sheet by the College Board

III. Programming
(Time: about 5 weeks)

A. Topics
   1. Using pseudocode and flowcharts to write computer programs
   2. Conditionals, loops, and lists in computer programming
   3. Using comments in computer programs
   4. Importance of documenting the development of a program

B. Resources
   1. Scratch programming language
   2. Snap! programming language
   3. Various resources from Code.org
   4. *Scratch Curriculum Guide* from ScratchED

C. Activities
   1. Create simple Scratch programs using conditionals and loops, having a flowchart or pseudocode for each program developed by the students
   2. Rock-Paper-Scissors game
   3. Collaborative program creation, with a shared development journal

D. Readings
   1. *Scratch Curriculum Guide Workbook*
   2. *Getting Started with Scratch* from MIT Media Lab

IV. Internet
(Time: about 2 weeks)

A. Topics
   1. How the Internet functions
   2. Packet switching
   3. Privacy and the Internet

B. Resources
   1. Various resources at Code.org
   2. *AP Computer Science Principles Internet Curriculum Module* from the College Board
   3. *Jot Thoughts* Activity from Cooperative Learning Resources
   4. *Days 8-9 Privacy* Activity from CS10KCommunity

C. Activities
   1. Role-playing activity for packet switching on the Internet
   2. Written summary of how the Internet works
   3. Students, in small groups, will make a presentation about some aspect of privacy and the Internet
   4. How the Internet and routers work activity

D. Readings
   1. *AP Computer Science Principles Internet Curriculum Module*, Lesson 1, Activity 1 and Lesson 2, Activity 2

V. Impact of Computing - Revisited
(Time: about 1.5 weeks)

A. Topics
   1. Innovation in computing

B. Resources
   1. Video editing software
   2. *Jot Thoughts* by Cooperative Learning Resources

C. Activities
   1. Research computing innovation and write a research paper
   2. Students, working in small groups, will develop videos that demonstrate features of computing innovation

D. Readings
   1. *Jot Thoughts*

VI. Creating Websites
(Time: about 3 weeks)

A. Topics
   1. Web development tool for creating website
   2. HTML and CSS

B. Resources
   1. Google for Education – Google Pages
   2. *HTML and JavaScript Basics*

C. Activities
   1. Use a web development tool to create a website
   2. Use HTML and CSS to build a website

D. Readings
   1. *HTML and JavaScript Basics* sections on HTML

VII. Impact of Computing - Revisited
(Time: about 1 week)

A. Topics
   1. Benefits and harm of computing innovation

B. Resources
   1. Various articles from the Internet from websites such as computer.org or news.mit.edu
   2. Video editing software

C. Activities
   1. Working in small groups, students will create a video about the benefits and harm of computing innovation
   2. Research paper on benefits and harm of computing innovation

D. Readings
   1. Various articles from the Internet from websites such as computer.org or news.mit.edu

VIII. Data and Abstraction
(Time: about 4 weeks)

A. Topics
   1. Computer images are composed of pixels
   2. Binary
   3. Computer hardware
   4. Operating systems and software
   5. Large data sets

B. Resources
   1. Various resources from Code.org
   2. *Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion*
   3. *Computes Bug Me! Creativity Lesson Plan* from CS10KCommunity
   4. *AP Computer Science Principles Data Abstraction and Procedural Abstraction Curriculum Module* from the College Board
   5. Datasets from Data.gov

C. Activities
   1. Student demonstration and oral report of the use of the pixilation tool to pixelate images
   2. Construct binary patterns using beads, colored paper or other materials
   3. Read about and write a summary of how bits are fundamental to everything
   4. Describe the components that make up a computer and are found on the motherboard
   5. Write a summary of the relationship between the different layers of software found on modern computers
   6. Analyze a large dataset and find patterns

D. Readings
   1. *Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion,* pages 1 to 18

IX. Impact of Computing - Revisited (Time: about 2 weeks)

A. Topics
   1. Creation of images
   2. Image manipulation
   3. Impact of computing innovation

B. Resources
   1. Photo editing software

C. Activities
   1. Create and manipulate images to illustrate the impact of computing innovations

D. Readings
   1. None

X. Programming - Revisited (Time: about 5 weeks)

A. Topics
   1. Introduction to JavaScript
   2. Creating programs in JavaScript

B. Resources
   1. *HTML and JavaScript Basics*
   2. Pencil Code at pencilcode.net

C. Activities
   1. Tutorial on JavaScript
   2. Construct JavaScript programs based on student interest

D. Readings
   1. *HTML and JavaScript Basics* – Sections on JavaScript

XI. AP Computer Science Principles Performance Tasks (Time: about 4 weeks)

A. Topics
   1. Explore – Impact of Computing
   2. Create – Applications from Ideas

B. Resources
   1. AP Computer Science Principles Curriculum Framework by the College Board

C. Activities
   1. Students will complete two AP Computer Science Principles performance tasks, creating a computing artifact and a writing component for each

D. Readings
   1. *AP Computer Science Principles Performance Guide*