

AP Mobile Computer Science Principles

Course Description

Mobile CSP is a project-based course based on the [AP CS Principles Framework](#). The course covers the **7 Big Ideas** and **6 Computational Thinking Practices**. During the course, students complete two collaborative programming projects and an individual research and writing project on the impact of a recent, computing innovation that appeals to the student. These projects conform to the College Board's two **performance tasks** on programming and impact. The emerging CS Principles AP course will use these performance tasks, in addition to a written exam, as a primary means for a student to demonstrate what they've learned.

Twenty-eight lessons and projects focus on [building socially useful mobile apps](#) with App Inventor for Android. Another 30 lessons focus on [computer science topics](#) ranging from algorithms to binary numbers to computer security. Readings from [Blown to Bits](#) ask students to reflect on some of the big societal issues that characterize 21st century computing, such as privacy, security, social networking.

AP CS Principles Exam

Students who complete this course will be prepared to take the AP CS Principles Exam. The [AP CS Principles Framework](#) is followed in conjunction with the official [Mobile CSP Syllabus](#).

Prerequisites (As described by the College Board)

It is recommended that a student in the AP Computer Science Principles course should have successfully completed a first-year high school algebra course with a strong foundation in basic algebraic concepts dealing with function notation, such as $f(x) = 5x^2$ and problem-solving strategies that require multiple approaches and collaborative efforts. In addition, students should be able to use a Cartesian (x, y) coordinate system to represent points on a plane. It is important that students and their advisers understand that any significant computer science course builds upon a foundation of mathematical reasoning that should be acquired before attempting such a course.

Programming Environment:

App Inventor for Android (ai2.appinventor.mit.edu), a free online software platform, is used in this course to build mobile apps for Android devices.

Online Resources:

The complete curriculum is hosted online and free of charge: <https://ram8647.appspot.com/mobileCSP>. The course uses many freely available resources that are only available online to ensure that the course material is current and adaptable. Students maintain individual online portfolios of their course work by using Google sites (<https://www.google.com/sites/overview.html>). Self-check and live coding exercises make use of Quizly (<https://github.com/ram8647/quizly>), a Web-based live coding platform for App Inventor. Throughout the course, students will also use a number of online articles and videos from sources such as The New York Times (www.nytimes.com), Wikipedia (www.wikipedia.org), CS Bits and Bytes (<http://www.nsf.gov/cise/csbytes/>), Logic.ly (www.logic.ly), YouTube (www.youtube.com), and CS Unplugged (<http://csunplugged.org>).

Reference Text:

- [App Inventor 2: Create Your Own Android Apps. David Wolber, Hal Abelson, Ellen Spertus, and Liz Looney O'Reilly Media, Inc., 2014](#)
- [Blown to Bits: Your Life, Liberty, and Happiness After the Digital Explosion. Hal Abelson, Ken Ledeen, Harry Lewis. Addison-Wesley, 2010](#)

AP Mobile Computer Science Principles

Pacing Guide		
Unit	Topic	Suggested Timing
Unit 1	Getting Started: Preview & Setup Introduction to Mobile Apps & Pair Programming Creating Graphics & Images Bit by Bit	approx. 8 weeks
Unit 2	Create Task: Programming Performance Task #1 Exploring Computing: Animation, Simulation, & Modeling Explore Task: Impact of Computing Innovations Performance Task #1	approx. 9 weeks
Unit 3	Algorithms and Procedural Abstractions Using and Analyzing Data & Information Explore Task: Impact of Computing Innovations Performance Task #2	approx. 9 weeks
Unit 4	Communication Through The Internet Create Task: Programming Performance Task #2 Data Project (Optional)	approx. 9 weeks

Educational Technology Standards

8.1.12.A.1, 8.1.12.A.2, 8.1.12.A.3, 8.1.12.B.2, 8.1.12.C.1, 8.1.12.D.1, 8.1.12.D.4, 8.1.12.D.5, 8.1.12.E.2, 8.1.12.F.1

➤ **Technology Operations and Concepts**

- Create a personal digital portfolio which reflects personal and academic interests, achievements, and career aspirations by using a variety of digital tools and resources.
- Produce and edit a multi-page digital document for a commercial or professional audience and present it to peers and/or professionals in that related area for review.
- Collaborate in online courses, learning communities, social networks or virtual worlds to discuss a resolution to a problem or issue.

➤ **Creativity and Innovation**

- Apply previous content knowledge by creating and piloting a digital learning game or tutorial.

➤ **Communication and Collaboration**

- Develop an innovative solution to a real world problem or issue in collaboration with peers and experts, and present ideas for feedback through social media or in an online community.

➤ **Digital Citizenship**

- Demonstrate appropriate application of copyright, fair use and/or Creative Commons to an original work.
- Research and understand the positive and negative impact of one's digital footprint.
- Analyze the capabilities and limitations of current and emerging technology resources and assess their potential to address personal, social, lifelong learning, and career needs.

➤ **Research and Information Literacy**

- Research and evaluate the impact on society of the unethical use of digital tools and present your research to peers.

➤ **Critical Thinking, Problem Solving, Decision Making**

- Evaluate the strengths and limitations of emerging technologies and their impact on educational, career, personal and or social needs.

Career Ready Practices

Career Ready Practices describe the career-ready skills that all educators in all content areas should seek to develop in their students. They are practices that have been linked to increase college, career, and life success. Career Ready Practices should be taught and reinforced in all career exploration and preparation programs with increasingly higher levels of complexity and expectation as a student advances through a program of study.

CRP1. Act as a responsible and contributing citizen and employee

Career-ready individuals understand the obligations and responsibilities of being a member of a community, and they demonstrate this understanding every day through their interactions with others. They are conscientious of the impacts of their decisions on others and the environment around them. They think about the near-term and long-term consequences of their actions and seek to act in ways that contribute to the betterment of their teams, families, community and workplace. They are reliable and consistent in going beyond the minimum expectation and in participating in activities that serve the greater good.

CRP2. Apply appropriate academic and technical skills.

Career-ready individuals readily access and use the knowledge and skills acquired through experience and education to be more productive. They make connections between abstract concepts with real-world applications, and they make correct insights about when it is appropriate to apply the use of an academic skill in a workplace situation.

CRP4. Communicate clearly and effectively and with reason.

Career-ready individuals communicate thoughts, ideas, and action plans with clarity, whether using written, verbal, and/or visual methods. They communicate in the workplace with clarity and purpose to make maximum use of their own and others' time. They are excellent writers; they master conventions, word choice, and organization, and use effective tone and presentation skills to articulate ideas. They are skilled at interacting with others; they are active listeners and speak clearly and with purpose. Career-ready individuals think about the audience for their communication and prepare accordingly to ensure the desired outcome.

CRP5. Consider the environmental, social and economic impacts of decisions.

Career-ready individuals understand the interrelated nature of their actions and regularly make decisions that positively impact and/or mitigate negative impact on other people, organization, and the environment. They are aware of and utilize new technologies, understandings, procedures, materials, and regulations affecting the nature of their work as it relates to the impact on the social

Career Ready Practices

condition, the environment and the profitability of the organization.

CRP6. Demonstrate creativity and innovation.

Career-ready individuals regularly think of ideas that solve problems in new and different ways, and they contribute those ideas in a useful and productive manner to improve their organization. They can consider unconventional ideas and suggestions as solutions to issues, tasks or problems, and they discern which ideas and suggestions will add greatest value. They seek new methods, practices, and ideas from a variety of sources and seek to apply those ideas to their own workplace. They take action on their ideas and understand how to bring innovation to an organization.

CRP7. Employ valid and reliable research strategies.

Career-ready individuals are discerning in accepting and using new information to make decisions, change practices or inform strategies. They use reliable research process to search for new information. They evaluate the validity of sources when considering the use and adoption of external information or practices in their workplace situation.

CRP8. Utilize critical thinking to make sense of problems and persevere in solving them.

Career-ready individuals readily recognize problems in the workplace, understand the nature of the problem, and devise effective plans to solve the problem. They are aware of problems when they occur and take action quickly to address the problem; they thoughtfully investigate the root cause of the problem prior to introducing solutions. They carefully consider the options to solve the problem. Once a solution is agreed upon, they follow through to ensure the problem is solved, whether through their own actions or the actions of others.

CRP9. Model integrity, ethical leadership and effective management.

Career-ready individuals consistently act in ways that align personal and community-held ideals and principles while employing strategies to positively influence others in the workplace. They have a clear understanding of integrity and act on this understanding in every decision. They use a variety of means to positively impact the directions and actions of a team or organization, and they apply insights into human behavior to change others' action, attitudes and/or beliefs. They recognize the near-term and long-term effects that management's actions and attitudes can have on productivity, morals and organizational culture.

CRP11. Use technology to enhance productivity.

Career Ready Practices

Career-ready individuals find and maximize the productive value of existing and new technology to accomplish workplace tasks and solve workplace problems. They are flexible and adaptive in acquiring new technology. They are proficient with ubiquitous technology applications. They understand the inherent risks-personal and organizational-of technology applications, and they take actions to prevent or mitigate these risks.

CRP12. Work productively in teams while using cultural global competence.

Career-ready individuals positively contribute to every team, whether formal or informal. They apply an awareness of cultural difference to avoid barriers to productive and positive interaction. They find ways to increase the engagement and contribution of all team members. They plan and facilitate effective team meetings.

Differentiated Instruction

Strategies to Accommodate Students Based on Individual Needs

<u>Time/General</u>	<u>Processing</u>	<u>Comprehension</u>	<u>Recall</u>
<ul style="list-style-type: none"> Extra time for assigned tasks Adjust length of assignment Timeline with due dates for reports and projects Communication system between home and school Provide lecture notes/outline 	<ul style="list-style-type: none"> Extra Response time Have students verbalize steps Repeat, clarify or reword directions Mini-breaks between tasks Provide a warning for transitions Reading partners 	<ul style="list-style-type: none"> Precise step-by-step directions Short manageable tasks Brief and concrete directions Provide immediate feedback Small group instruction Emphasize multi-sensory learning 	<ul style="list-style-type: none"> Teacher-made checklist Use visual graphic organizers Reference resources to promote independence Visual and verbal reminders Graphic organizers
<u>Assistive Technology</u>	<u>Tests/Quizzes/Grading</u>	<u>Behavior/Attention</u>	<u>Organization</u>
<ul style="list-style-type: none"> Computer/whiteboard Tape recorder Spell-checker Audio-taped books 	<ul style="list-style-type: none"> Extended time Study guides Shortened tests Read directions aloud 	<ul style="list-style-type: none"> Consistent daily structured routine Simple and clear classroom rules Frequent feedback 	<ul style="list-style-type: none"> Individual daily planner Display a written agenda Note-taking assistance Color code materials

Enrichment

Strategies Used to Accommodate Based on Students Individual Needs:

- Adaption of Material and Requirements
- Evaluate Vocabulary
- Additional Projects
- Independent Student Options
- Projects completed individual or with Partners
- Self-Selection of Research
- Tiered/Multilevel Activities
- Learning Centers
- Individual Response Board
- Independent Book Studies
- Open-ended activities
- Community/Subject expert mentorships

Assessments

Suggested Formative/Summative Classroom Assessments

Portfolios: In this course students will document their work on their **portfolios**. That is, they will post answers to reading questions, write-ups of hands-on tutorials, written responses to assigned readings, and documentation of creative programming projects on their personal portfolio page. Each student will create a portfolio using Google sites (<https://www.google.com/sites/overview.html>). The portfolios will promote collaboration and sharing -- students can learn from each other -- and will constitute a full record of what the students have done in the course that they can refer back to during and after the course and share with their friends and family. Portfolios will be graded periodically throughout the duration of the course.

Reading and Homework Assignments: There will be regular reading and/or out-of-class homework assignments. These may include reading a chapter from the textbook and/or completing a tutorial or worksheet. Brief, clear, and concise written responses to the study questions must be posted on students' portfolios.

Labs: This course will be taught in a computer lab. Students will have access to computers and mobile devices and any other necessary hardware, both during the class and during free periods. Students can work in the lab during their free periods. Internet access will be available to students throughout the course. In each unit, there will be at least three labs designed to practice and/or reinforce key concepts. Some are unplugged and others are completed in an online development environment. Most are completed in App Inventor.

Projects There will be two (2) creative programming projects in which students will use lab time to work both individually and collaboratively (in pairs) to create a socially useful mobile app that they propose (pitch), design, and implement. One of these will be a practice for the College Board's Create Performance Task. The second will be the official College Board Create Performance Task. Twelve (12) hours of class time will be provided for completion of the official Create Performance Task.

There will also be two (2) written research projects that students will work on individually. These research projects will focus on examining a computing innovation that has impacted society. One will be a practice for the Explore Performance Task. The second will be the College Board's Explore Performance Task. Eight (8) hours of class time will be provided for completion of the official College Board Explore Performance Task.

Oral and Video presentations: There will be approximately three (3) oral and/or videotaped presentations of students' projects during the course.

Quizzes and exams: There will be periodic quizzes, typically to wrap up the end of each unit, and a midterm exam given during the course. There will be a comprehensive final exam. Quizzes will be hand written and/or electronic and exams will be electronic.

Self-check and Live coding exercises: All lessons in this course are accompanied by short, interactive, self-check exercises that consist of multiple choice and fill-in question as well as automatically graded, live-coding, programming exercises (<https://github.com/ram8647/quizly>). These assessments are considered an essential part of the learning process. These are hosted online and may be done individually or with the class as a whole. Each question or exercise includes detailed feedback and students may repeat the question or exercise until it is correct.

AP Computer Science Principles Exam (AP Exam May 5, 2017) Students who complete this course will be prepared to take the AP CS Principles Exam.

Interdisciplinary Connections

English Language Arts

- Journal writing
- Close reading of industry-related content
- Create a brochure for a specific industry
- Keep a running word wall of industry vocabulary

Social Studies

- Research the history of a given industry/profession
- Research prominent historical individuals in a given industry/profession
- Use historical references to solve problems

World Language

- Translate industry-content
- Create a translated index of industry vocabulary
- Generate a translated list of words and phrases related to information technology

Math

- Compare and contrast use of equations and variables in algebra and programming.
- Program graphics and use the properties of geometric shapes
- Compare the computer graphic coordinate system with the Cartesian coordinate plane in math
- Compare probability and the use of random numbers in computer programming.
- Track and track various data, such as industry's impact on the GDP, career opportunities or among of individuals currently occupying careers

Fine & Performing Arts

- Create a poster recruiting young people to focus their studies on a career in Information Technology

Science

- Research the environmental impact of a given career or industry
- Research latest developments in Information technology
- Investigate applicable-careers in STEM fields

[New Jersey Student Learning Standards](#)

8.2 Technology Education, Engineering, Design, and Computational Thinking - Programming

Technology and Society

- 8.2.12.B.3: Analyze ethical and unethical practices around intellectual property rights as influenced by human wants and/or needs.

Design

- 8.2.12.C.1: Explain how open source technologies follow the design process.

Computational Thinking: Programming

- 8.2.12.E.1: Demonstrate an understanding of the problem-solving capacity of computers in our world.
- 8.2.12.E.2: Analyze the relationships between internal and external computer components.
- 8.2.12.E.3: Use a programming language to solve problems or accomplish a task.
- 8.2.12.E.4: Use appropriate terms in conversation.

9.3– Career and Technical Education

Career Cluster: Information Technology (IT)

- 9.3.12.IT.11: Demonstrate knowledge of the hardware components associated with information systems.
- 9.3.12.IT-SUP.9: Employ technical writing and documentation skills in support of an information system.

Pathway: Programming & Software Development (IT-PRG)

- 9.3.12.IT-PRG.4: Demonstrate the effective use of software development tools to develop software applications.
- 9.3.12.IT-PRG.5: Apply an appropriate software development process to design a software application.
- 9.3.12.IT-PRG.6: Program a computer application using the appropriate programming language.
- 9.3.12.IT-PRG.7: Demonstrate software testing procedures to ensure quality products.

Common Career Technical Core (CCTC) **Career Cluster Information Technology**

IT.11 – Demonstrate knowledge of the hardware components associated with information systems.

- IT.11.1 - None available at this time.

IT-SUP.9 - Employ technical writing and documentation skills in support of an information system.

- IT-SUP.9.3 - Design technical documentation.

IT-PRG.4 - Demonstrate the effective use of software development tools to develop software applications.

- IT-PRG.4.1 - Employ tools in developing software applications.
- IT-PRG.4.3 - Apply language-specific programming tools/techniques.

IT-PRG.5 - Apply an appropriate software development process to design a software application.

- IT-PRG.5.1 - Describe software development processes and methodology.

IT-PRG.6 - Program a computer application using the appropriate programming language.

- IT-PRG.6.1 - Explain programming language concepts.
- IT-PRG.6.2 - Summarize program development methodology.
- IT-PRG.6.3 - Demonstrate proficiency in developing an application using an appropriate programming language.
- IT-PRG.6.4 - Explain basic software systems implementation.
- IT-PRG.6.6 - Resolve problems with integration.

IT-PRG.7 - Demonstrate software testing procedures to ensure quality products.

- IT-PRG.7.1 - Develop a software test plan.

Common Core State Standards (CCSS)

CCSS - English-Language Arts

Key Ideas and Details:

- CCSS.ELA-LITERACY.RL.11-12.1 Cite strong and thorough textual evidence to support analysis of what the text says explicitly as well as inferences drawn from the text, including determining where the text leaves matters uncertain.

Integration of Knowledge and Ideas:

- CCSS.ELA-LITERACY.W.11-12.1 Write arguments to support claims in an analysis of substantive topics or texts, using valid reasoning and relevant and sufficient evidence.

Production and Distribution of Writing:

- CCSS.ELA-LITERACY.W.11-12.4 Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.

Research to Build and Present Knowledge:

- CCSS.ELA-LITERACY.W.11-12.7 Conduct short as well as more sustained research projects to answer a question (including a self-generated question) or solve a problem; narrow or broaden the inquiry when appropriate; synthesize multiple sources on the subject, demonstrating understanding of the subject under investigation.

Range of Writing:

- CCSS.ELA-LITERACY.W.11-12.10 Write routinely over extended time frames (time for research, reflection, and revision) and shorter time frames (a single sitting or a day or two) for a range of tasks, purposes, and audiences

Common Core State Standards (CCSS)

CCSS - Mathematics

Reason quantitatively and use units to solve problems:

- CCSS.MATH.CONTENT.HSN-Q.A.2 Define appropriate quantities for the purpose of descriptive modeling.

Create equations that describe numbers or relationships:

- CCSS.MATH.CONTENT.HSA-CED.A.1 Create equations and inequalities in one variable and use them to solve problems.

Analyze functions using different representations:

- CCSS.MATH.CONTENT.HSF-IF.C.7 Graph functions expressed symbolically and show key features of the graph.

Apply geometric concepts in modeling situations:

- CCSS.MATH.CONTENT.HSG-MG.A.1 Use geometric shapes, their measures, and their properties to describe objects

Calculate expected values and use them to solve problems:

- CCSS.MATH.CONTENT.HSS-MD.A.1 Define a random variable for a quantity of interest by assigning a numerical value to each event in a sample space;
- CCSS.MATH.CONTENT.HSS-MD.A.2 Calculate the expected value of a random variable;

Practice Standards - Mathematics

- 1. Make sense of problems and persevere in solving them.
- 2. Reason abstractly and quantitatively.
- 3. Construct viable arguments and critique the reasoning of others.
- 4. Model with mathematics.
- 5. Use appropriate tools strategically.
- 6. Attend to precision.
- 7. Look for and make use of structure.
- 8. Look for and express regularity in repeated reasoning.

<p>Course: AP Mobile Computer Science Principles</p> <p>Unit: 4 – Communication Through The Internet</p> <p>Create Task: Programming Performance Task #2</p> <p>Data Project</p> <p>Grade Level: 9-12</p>	<p>Unit Overview:</p> <p>Unit 4 focuses on the Internet, one of the big ideas in computer science. The App Inventor lessons in this unit show different ways to use the internet in apps, including the ability to send text messages over Wifi, finding directions via the Google Maps API. The CS Principles lessons focus on the Internet, how it works, how it enables innovation and collaboration, and security concerns for using it.</p> <p>The Create Task: Programming Performance Task #2 requires students to develop a program on a topic that interests them or solves a problem. It is strongly recommended that a portion of the program involve some form of collaboration with another student in the class. The program development must also involve a significant amount of independent work in the planning and design parts of the process. Students and teachers must carefully follow all of the directions as presented by the College Board and will need to address the requirements and prompts as outlined in the Performance Task directions. (AP Computer Science Principles Course and Exam Description pages 76-81)</p> <p>(Optional) Students will work with a partner to identify and analyze a large data set that of interest. A collaborative project in the form of a PowerPoint presentation and a portfolio write up on the investigative process will be required. Students will also complete individual write-ups to post on the students’ portfolio.</p>
<p>New Jersey Student Learning Standards (NJSLS): 8.2.12.B.3, 8.2.12.C.1, 8.2.12.E.1, 8.2.12.E.2, 8.2.12.E.3, 8.2.12.E.4 9.3.12.IT.11, 9.3.12.IT-SUP.9.3, 9.3.12.IT-PRG.4.1, 9.3.12.IT-PRG.4.3, 9.3.12.IT-PRG.5.1, 9.3.12.IT-PRG.6.1, 9.3.12.IT-PRG.6.2, 9.3.12.IT-PRG.6.3, 9.3.12.IT-PRG.6.4, 9.3.12.IT-PRG.6.6, 9.3.12.IT-PRG.7.1</p>	
<p>Common Career Technical Core (CCTC): IT.11, IT-SUP.9.3, IT-PRG.4.1, IT-PRG.4.3, IT-PRG.5.1, IT-PRG.6.1, IT-PRG.6.2, IT-PRG.6.3, IT-PRG.6.4, IT-PRG.6.6, IT-PRG.7.1</p>	
<p>Common Core State Standards (CCSS): RL.11-12.1; W.11-12.1; W.11-12.4; W.11-12.7; W.11-12.10;</p>	
<p>LINK TO CONTENT STANDARDS: https://secure-media.collegeboard.org/digitalServices/pdf/ap/ap-computer-science-principles-course-and-exam-</p>	

[description.pdf](#)

Big Ideas Standard:

- BI 1 – Computing is a creative activity.
- BI 5 – Programming enables problem solving, human expression, and creation of knowledge.
- BI 6 – The Internet pervades modern computing.
- BI 7 – Computing has global Impact

Computational Thinking Practices:

- P1 - Study the effects and they learn to draw connections between different computing concepts.
- P2 - Creating Computational Artifacts - engage in the creative aspects of computing by designing and developing interesting computational artifacts as well as by applying computing techniques to creatively solve problems.
- P3 – Abstracting - use abstraction to develop models and simulations of natural and artificial phenomena, use them to make predictions about the world, and analyze their efficacy and validity.
- P4 - Analyzing Problems and Artifacts - design and produce solutions, models, and artifacts, and they evaluate and analyze their own computational work as well as the computational work others have produced.
- P5 – Communicating - describe computation and the impact of technology and computation, explain and justify the design and appropriateness of their computational choices, and analyze and describe both computational artifacts and the results or behaviors of such artifacts.
- P6 – Collaborating - collaborate on a number of activities, including investigation of questions using data sets and in the production of computational artifacts.

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>ESSENTIAL KNOWLEDGE, SKILLS, AND ENDURING UNDERSTANDINGS:</p> <p>A: STUDENTS WILL KNOW: EU 1.2 Computing enables people to use creative development processes when using computing tools and techniques to create computational artifacts for creative expression of ideas or to solve a problem. EU 5.3 Programming is facilitated by appropriate abstractions. EU 6.1 The Internet is a network of autonomous systems.</p>	<p>ESSENTIAL QUESTIONS THAT WILL FOCUS TEACHING AND LEARNING:</p> <p>B11 - How can a creative development process affect the creation of computational artifacts?</p> <p>B11 - How can computing and the use of computational tools foster creative expression?</p>	<p>C: STUDENTS WILL BE ABLE TO: LO 1.2.2 Create a computational artifact using computing tools and techniques to solve a problem. [P2] LO 1.2.4 Collaborate in the creation of computational artifacts. [P6] LO 5.3.1 Use abstraction to</p>	<ul style="list-style-type: none"> • What is the Internet? • No Texting While Busy Tutorial, • Cloud Computing and Ethics, • How the Internet Works, • My Directions Tutorial, • Cryptography Basics, 	<p>Unit 7 - Communication Through The Internet Instructor Materials</p> <p>Unit 7 - Communication Through The Internet</p>

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EU 6.2 Characteristics of the Internet influence the systems built on it. EU 6.3 Cybersecurity is an important concern for the Internet and the systems built on it. EU 7.1 Computing enhances communication, interaction, and cognition. EU 7.3 Computing has a global affect - both beneficial and harmful - on people and society. EU 7.4 Computing innovations influence and are influenced by the economic, social, and cultural contexts in which they are designed and used.</p> <p>B: STUDENTS WILL UNDERSTAND THAT: EK 1.2.2A Computing tools and techniques can enhance the process of finding a solution to a problem. EK 1.2.4B Effective collaborative teams consider the use of online collaborative tools. EK 5.3.1M Application program interfaces (APIs) and libraries simplify complex programming tasks. EK 5.3.1N Documentation for an API/library is an important aspect of programming. EK 5.3.1O APIs connect software components, allowing them to communicate. EK 6.1.1A The Internet connects devices and networks all over the world. EK 6.1.1B An end-to-end architecture facilitates connecting new devices and networks on the Internet. EK 6.1.1C Devices and networks that make up the Internet are connected and communicate using addresses and protocols.</p>	<p>BI1 - How can computing extend traditional forms of human expression and experience?</p> <p>BI6 – What is the internet? How is it built? How does it function?</p> <p>BI6 – What aspect of the Internet’s design and development have helped it scale and flourish?</p> <p>BI6 – How is cybersecurity impacting the ever-increasing number of Internet users?</p> <p>BI5 - How are programs developed to help people, organizations, or society solve problems?</p> <p>BI5 - How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?</p> <p>BI5 - How do computer programs implement algorithms?</p> <p>BI5 - How does abstraction make the development of computer programs possible?</p> <p>BI5 - How do people develop</p>	<p>manage complexity in programs. [P3] LO 6.1.1 Explain the abstractions in the Internet and how the Internet functions. [P3] EXCLUSION STATEMENT (for LO 6.1.1): Specific devices used to implement the abstractions in the Internet are beyond the scope of this course and the AP Exam. LO 6.2.1 Explain characteristics of the Internet and the systems built on it. [P5] LO 6.2.2 Explain how the characteristics of the Internet influence the systems built on it. [P4] LO 6.3.1 Identify existing cybersecurity concerns and potential options to address these issues with the Internet and the systems built on it. [P1] LO 7.1.1 Explain how computing innovations affect communication, interaction, and cognition. [P4]</p>	<ul style="list-style-type: none"> • Cryptography: Securing the Internet • Blown to Bits: Cryptography, • Socially Aware App: Broadcast Hub, • Wrap up 	

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EK 6.1.1D The Internet and the systems built on it facilitate collaboration.</p> <p>EK 6.1.1E Connecting new devices to the Internet is enabled</p> <p>EK 6.1.1F The Internet is built on evolving standards, including those for addresses and names.</p> <p style="padding-left: 40px;">EXCLUSION STATEMENT (for EK 6.1.1F): Specific details of any particular standard for addresses are beyond the scope of this course and the AP Exam.</p> <p>EK 6.1.1G The domain name system (DNS) translates names to IP addresses.</p> <p>EK 6.1.1H The number of devices that could use an IP address has grown so fast that a new protocol (IPv6) has been established to handle routing of many more devices.</p> <p>EK 6.1.1I Standards such as hypertext transfer protocol (HTTP), IP, and simple mail transfer protocol (SMTP) are developed and overseen by the Internet Engineering Task Force (IETF).</p> <p>EK 6.2.1A The Internet and the systems built on it are hierarchical and redundant.</p> <p>EK 6.2.1B The domain name syntax is hierarchical. EK 6.2.1C IP addresses are hierarchical.</p> <p>EK 6.2.1D Routing on the Internet is fault tolerant and redundant.</p> <p>EK 6.2.2A Hierarchy and redundancy help systems scale.</p> <p>EK 6.2.2B The redundancy of routing (i.e., more than one way to route data) between two points on the Internet increases the reliability of the Internet and helps it scale to more devices and</p>	<p>and test computer programs?</p> <p>BI5 - Which mathematical and logical concepts are fundamental to computer programming?</p> <p>BI7 - How does computing enhance human communication, interaction, and cognition?</p> <p>BI7 - How does computing enable innovation?</p> <p>BI7 - What are some potential beneficial and harmful effects of computing?</p> <p>GUIDING QUESTIONS:</p> <p>What is the Internet, how is it built, and how does it function?</p> <p>What aspects of the Internet's design and development have helped it scale and flourish?</p> <p>How is cybersecurity impacting the ever increasing number of Internet users?</p>	<p>LO 7.1.2 Explain how people participate in a problem solving process that scales. [P4]</p> <p>LO 7.3.1 Analyze the beneficial and harmful effects of computing. [P4]</p> <p>LO 7.4.1 Explain the connections between computing and economic, social, and cultural contexts. [P1]</p>		

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>more people. EK 6.2.2C Hierarchy in the DNS helps that system scale. EK 6.2.2D Interfaces and protocols enable widespread use of the Internet. EK 6.2.2E Open standards fuel the growth of the Internet. EK 6.2.2F The Internet is a packet-switched system through which digital data is sent by breaking the data into blocks of bits called packets, which contain both the data being transmitted and control information for routing the data. EXCLUSION STATEMENT (for EK 6.2.2F): Specific details of any particular packet-switching system are beyond the scope of this course and the AP Exam. EK 6.2.2H Standards for sharing information and communicating between browsers and servers on the Web include HTTP and secure sockets layer/transport layer security (SSL/TLS). EXCLUSION STATEMENT (for EK 6.2.2H): Understanding the technical aspects of how SSL/TLS works is beyond the scope of this course and the AP Exam. EK 6.2.2J The bandwidth of a system is a measure of bit rate—the amount of data (measured in bits) that can be sent in a fixed amount of time. EK 6.2.2K The latency of a system is the time elapsed between the transmission and the receipt of a request. EK 6.3.1A The trust model of the Internet involves trade-offs.</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EK 6.3.1B The DNS was not designed to be completely secure.</p> <p>EK 6.3.1C Implementing cybersecurity has software, hardware, and human components.</p> <p>EK 6.3.1D Cyber warfare and cybercrime have widespread and potentially devastating effects.</p> <p>EK 6.3.1E Distributed denial-of-service attacks (DDoS) compromise a target by flooding it with requests from multiple systems.</p> <p>EK 6.3.1F Phishing, viruses, and other attacks have human and software components.</p> <p>EK 6.3.1G Antivirus software and firewalls can help prevent unauthorized access to private data.</p> <p>EK 6.3.1H Cryptography is essential to many models of cybersecurity.</p> <p>EK 6.3.1I Cryptography has a mathematical foundation.</p> <p style="padding-left: 20px;">EXCLUSION STATEMENT (for EK 6.3.1I): Specific mathematical functions used in cryptography are beyond the scope of this course and the AP Exam.</p> <p>EK 6.3.1J Open standards help ensure cryptography is secure.</p> <p>EK 6.3.1K Symmetric encryption is a method of encryption involving one key for encryption and decryption.</p> <p style="padding-left: 20px;">EXCLUSION STATEMENT (for EK 6.3.1K): The methods used in encryption are beyond the scope of this course and the AP Exam.</p> <p>EK 6.3.1L Public key encryption, which is not symmetric, is an encryption method that is widely used because of the functionality it provides.</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EXCLUSION STATEMENT (for EK 6.3.1L): The mathematical methods used in public key cryptography are beyond the scope of this course and the AP Exam.</p> <p>EK 6.3.1M Certificate authorities (CAs) issue digital certificates that validate the ownership of encrypted keys used in secured communications and are based on a trust model.</p> <p>EXCLUSION STATEMENT (for EK 6.3.1M): The technical details of the process CAs follow are beyond the scope of this course and the AP Exam.</p> <p>EK 7.1.1A Email, SMS, and chat have fostered new ways to communicate and collaborate.</p> <p>EK 7.1.1D Cloud computing fosters new ways to communicate and collaborate.</p> <p>EK 7.1.1I Global Positioning System (GPS) and related technologies have changed how humans travel, navigate, and find information related to geolocation.</p> <p>EK 7.1.1J Sensor networks facilitate new ways of interacting with the environment and with physical systems.</p> <p>EK 7.1.1K Smart grids, smart buildings, and smart transportation are changing and facilitating human capabilities.</p> <p>EK 7.1.1L Computing contributes to many assistive technologies that enhance human capabilities.</p> <p>EK 7.1.2F Crowdsourcing offers new models for collaboration, such as connecting people with jobs and</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>businesses with funding. EK 7.3.1A Innovations enabled by computing raise legal and ethical concerns. EK 7.3.1B Commercial access to music and movie downloads and streaming raises legal and ethical concerns. EK 7.4.1B Mobile, wireless, and networked computing have an impact on innovation throughout the world.</p>				
<p>A: STUDENTS WILL KNOW: EK 1.1.1A A creative process in the development of a computational artifact can include, but is not limited to, employing nontraditional, non-prescribed techniques; the use of novel combinations of artifacts, tools, and techniques; and the exploration of personal curiosities. EK 1.1.1B Creating computational artifacts employs an iterative and often exploratory process to translate ideas into tangible form. EK 1.2.1B Creating computational artifacts requires understanding of and use of software tools and services. EK 1.2.1C Computing tools and techniques are used to create computational artifacts and can include, but are not limited to, programming integrated development environments (IDEs), spreadsheets, 3D printers, or text editors. EK 1.2.1D A creatively developed computational artifact can be created by using nontraditional, non-prescribed computing techniques. EK 1.2.1E Creative expressions in a computational artifact can reflect</p>	<p>BI1 - How can a creative development process affect the creation of computational artifacts? BI1 - How can computing and the use of computational tools foster creative expression? BI1 - How can computing extend traditional forms of human expression and experience? BI2 – How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer? BI2 – How does abstraction help us in writing programs, creating computational artifacts and solving problems?</p>	<p>STUDENTS WILL BE ABLE TO: LO 1.1.1 Apply a creative development process when creating computational artifacts. [P2] LO 1.2.1 Create a computational artifact for creative expression. [P2] LO 1.2.2 Create a computational artifact using computing tools and techniques to solve a problem. [P2] LO 1.2.3 Create a new computational artifact by combining or modifying existing artifacts. [P2] LO 1.2.4 Collaborate in the creation of computational artifacts. [P6]</p>	<p>ASSESSMENT (EVIDENCE OF KNOWLEDGE AND UNDERSTANDING)</p> <p>STUDENTS WILL:</p> <ul style="list-style-type: none"> • Create an application that can be used on an Android device. • Complete an official performance task (turned in to the College Board in April) 	<p>Create: Programming Performance Task #2 Create: Programming Performance Task #2 - INSTRUCTOR MATERIALS CSP Performance Tasks Data Project (Optional) Data Project Instructor Materials (Optional) Exam 2 - Final Exam 2 Instructor Materials - Final Unit 8 - AP CS Principles</p>

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>personal expressions of ideas or interests.</p> <p>EK 1.2.2B A creative development process for creating computational artifacts can be used to solve problems when traditional or prescribed computing techniques are not effective.</p> <p>EK 1.2.3A Creating computational artifacts can be done by combining and modifying existing artifacts or by creating new artifacts.</p> <p>EK 1.2.3B Computation facilitates the creation and modification of computational artifacts with enhanced detail and precision.</p> <p>EK 1.2.3C Combining or modifying existing artifacts can show personal expression of ideas.</p> <p>EK 1.2.4A A collaboratively created computational artifact reflects effort by more than one person.</p> <p>EK 1.2.4C Effective collaborative teams practice interpersonal communication, consensus building, conflict resolution, and negotiation.</p> <p>EK 1.2.4D Effective collaboration strategies enhance performance.</p> <p>EK 1.2.4E Collaboration facilitates the application of multiple perspectives (including sociocultural perspectives) and diverse talents and skills in developing computational artifacts.</p> <p>EK 1.2.4F A collaboratively created computational artifact can reflect personal expressions of ideas.</p> <p>EK 2.2.1A The process of developing an abstraction involves removing detail and generalizing functionality.</p> <p>EK 2.2.1B An abstraction extracts common features from specific examples in order to generalize concepts.</p>	<p>BI2 – How can computational models and simulations help generate new understanding and knowledge?</p> <p>BI4 - How are algorithms implemented and executed on computers and computational devices?</p> <p>BI4 - Why are some languages better than others when used to implement algorithms?</p> <p>BI 4 - What kinds of problems are easy, what kinds are difficult, and what kinds are impossible to solve algorithmically?</p> <p>BI4 - How are algorithms evaluated?</p> <p>BI5 - How are programs developed to help people, organizations, or society solve problems?</p> <p>BI5 - How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?</p> <p>BI5 - How do computer programs implement algorithms?</p> <p>BI5 - How does abstraction</p>	<p>LO 1.2.5 Analyze the correctness, usability, functionality, and suitability of computational artifacts. [P4]</p> <p>LO 2.2.1 Develop an abstraction when writing a program or creating other computational artifacts. [P2]</p> <p>LO 2.2.2 Use multiple levels of abstraction to write programs. [P3]</p> <p>LO 4.1.1 Develop an algorithm for implementation in a program. [P2]</p> <p>LO 4.1.2 Express an algorithm in a language. [P5]</p> <p>LO 5.1.1 Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge. [P2]</p> <p>LO 5.1.2 Develop a correct program to solve problems. [P2]</p> <p>LO 5.1.3 Collaborate to develop a program. [P6]</p> <p>LO 5.2.1 Explain how programs implement algorithms. [P3]</p>		<p>Exam Prep</p> <p>Unit 8 - AP CS Principles Exam Prep Instructor Materials</p>

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EK 2.2.1C An abstraction generalizes functionality with input parameters that allow software reuse. EXCLUSION STATEMENT (for EK 2.2.1C): An understanding of the difference between value and reference parameters is beyond the scope of this course and the AP Exam. EK 2.2.2A Software is developed using multiple levels of abstractions, such as constants, expressions, statements, procedures, and libraries. EK 2.2.2B Being aware of and using multiple levels of abstraction in developing programs helps to more effectively apply available resources and tools to solve problems. EK 4.1.1A Sequencing, selection, and iteration are building blocks of algorithms. EK 4.1.1B Sequencing is the application of each step of an algorithm in the order in which the statements are given. EK 4.1.1D Iteration is the repetition of part of an algorithm until a condition is met or for a specified number of times. EK 4.1.1E Algorithms can be combined to make new algorithms. EK 4.1.1G Knowledge of standard algorithms can help in constructing new algorithms. EK 4.1.1I Developing a new algorithm to solve a problem can yield insight into the problem. EK 4.1.2A Languages for algorithms include natural language, pseudocode, and visual and textual programming languages. EK 4.1.2B Natural language and pseudocode describe algorithms so that humans can understand them.</p>	<p>make the development of computer programs possible? B15 - How do people develop and test computer programs? B15 - Which mathematical and logical concepts are fundamental to computer programming?</p> <p>GUIDING QUESTIONS:</p> <p>What kind of app would you and your partner like to create? What programming language will you use to create this app? What do you want your app to do? What is the purpose/function of the app you will create?</p>	<p>LO 5.3.1 Use abstraction to manage complexity in programs. [P3] LO 5.4.1 Evaluate the correctness of a program. [P4] LO 5.5.1 Employ appropriate mathematical and logical concepts in programming. [P1]</p>		

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EK 4.1.2C Algorithms described in programming languages can be executed on a computer.</p> <p>EK 4.1.2D Different languages are better suited for expressing different algorithms.</p> <p>EK 5.1.1A Programs are developed and used in a variety of ways by a wide range of people depending on the goals of the programmer.</p> <p>EK 5.1.1B Programs developed for creative expression, to satisfy personal curiosity, or to create new knowledge may have visual, audible, or tactile inputs and outputs.</p> <p>EK 5.1.1C Programs developed for creative expression, to satisfy personal curiosity, or to create new knowledge may be developed with different standards or methods than programs developed for widespread distribution.</p> <p>EK 5.1.1D Additional desired outcomes may be realized independently of the original purpose of the program.</p> <p>EK 5.1.1E A computer program or the results of running a program may be rapidly shared with a large number of users and can have widespread impact on individuals, organizations, and society.</p> <p>EK 5.1.2A An iterative process of program development helps in developing a correct program to solve problems.</p> <p>EK 5.1.2B Developing correct program components and then combining them helps in creating correct programs.</p> <p>EK 5.1.2C Incrementally adding tested program segments to correct working programs helps create large correct programs.</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>EK 5.1.2D Program documentation helps programmers develop and maintain correct programs to efficiently solve problems.</p> <p>EK 5.1.2E Documentation about program components, such as blocks and procedures, helps in developing and maintaining programs.</p> <p>EK 5.1.2F Documentation helps in developing and maintaining programs when working individually or in collaborative programming environments.</p> <p>EK 5.1.2G Program development includes identifying programmer and user concerns that affect the solution to problems.</p> <p>EK 5.1.2J A programmer designs, implements, tests, debugs, and maintains programs when solving problems.</p> <p>EK 5.1.3A Collaboration can decrease the size and complexity of tasks required of individual programmers.</p> <p>EK 5.1.3B Collaboration facilitates multiple perspectives in developing ideas for solving problems by programming.</p> <p>EK 5.1.3C Collaboration in the iterative development of a program requires different skills than developing a program alone.</p> <p>EK 5.1.3D Collaboration can make it easier to find and correct errors when developing programs.</p> <p>EK 5.1.3E Collaboration facilitates developing program components independently.</p> <p>EK 5.1.3F Effective communication between participants is required for successful collaboration when developing programs.</p> <p>EK 5.2.1A Algorithms are</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>implemented using program instructions that are processed during program execution. EK 5.2.1B Program instructions are executed sequentially. EK 5.2.1C Program instructions may involve variables that are initialized and updated, read, and written. EK 5.2.1D An understanding of instruction processing and program execution is useful for programming. EK 5.3.1A Procedures are reusable programming abstractions. EK 5.3.1B A procedure is a named grouping of programming instructions. EK 5.3.1C Procedures reduce the complexity of writing and maintaining programs. EK 5.3.1D Procedures have names and may have parameters and return values. EK 5.4.1A Program style can affect the determination of program correctness. EK 5.4.1C Meaningful names for variables and procedures help people better understand programs. EK 5.4.1D Longer code blocks are harder to reason about than shorter code blocks in a program. EK 5.4.1E Locating and correcting errors in a program is called debugging the program. EK 5.4.1F Knowledge of what a program is supposed to do is required in order to find most program errors. EK 5.5.1A Numbers and numerical concepts are fundamental to programming. EK 5.5.1B Integers may be constrained in the maximum and minimum values that can be represented in a program because of</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
<p>storage limitations. EXCLUSION STATEMENT (for EK 5.5.1B): Specific range limitations of all programming languages are beyond the scope of this course and the AP Exam. EK 5.5.1C Real numbers are approximated by floating-point representations that do not necessarily have infinite precision. EXCLUSION STATEMENT (for EK 5.5.1C): Specific sets of values that cannot be exactly represented by floating point numbers are beyond the scope of this course and the AP Exam. EK 5.5.1D Mathematical expressions using arithmetic operators are part of most programming languages. EK 5.5.1F Compound expressions using and, or, and not are part of most programming languages.</p> <p><i>B: STUDENTS WILL UNDERSTAND THAT:</i> EU 1.1 Creative development can be an essential process for creating computational artifacts. EU 1.2 Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem. EU 2.2 Multiple levels of abstraction are used to write programs or create other computational artifacts. EU 4.1 Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages. EU 5.1 Programs can be developed for creative expression, to satisfy</p>				

Student Learning Objectives (SLOs)	Essential Questions	Skills & Indicators	Sample Activities	Resources
personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society). EU 5.2 People write programs to execute algorithms. EU 5.3 Programming is facilitated by appropriate abstractions. EU 5.4 Programs are developed, maintained, and used by people for different purposes. EU 5.5 Programming uses mathematical and logical concepts.				

Unit 4 Vocabulary

Structured data
 Persistent data
 Shareable data
 Fusion table
 Database
 Copyright
 Big data
 Data mining
 Synchronize
 Open access
 Creative commons
 Digital Millennium Copyright Act (DMCA)
 Refactoring

Blown to Bits Chapter 6 Vocabulary

centralized systems
 commons
 DRAM
 DRM
 flooding
 gigabyte
 peer-to-peer architecture
 piracy
 sealed storage
 TPM

Blown to Bits Chapter 5 Vocabulary

AES
 certification authority
 cipher text
 DES
 decryption
 encryption
 packet
 plain text
 router

Suggested Unit Projects

Choose At Least One

Create: Programming Performance Task #2

<https://ram8647.appspot.com/mobileCSP/assessment?name=118>

Data Project

<https://ram8647.appspot.com/mobileCSP/assessment?name=124>

Suggested Structured Learning Experiences

Amazon Fulfillment Center Tour, Middletown, DE

<http://amazonfctours.com/>

Virtual Tour of Google's Oregon Data Center (10 Minutes)

<https://www.engadget.com/2016/03/24/google-360-video-tour-data-center/>

<http://www.google.com/about/datacenters/inside/streetview/>