DELAWARE VALLEY SCHOOL DISTRICT

PLANNED INSTRUCTION

A PLANNED COURSE FOR:

AP Computer Science Principles

Grade Level: 9th - 12th

Date of Board Approval: _____2018_____

DELAWARE VALLEY SCHOOL DISTRICT

Planned Instruction

Title of Planned Instruction: AP Computer Science Principles

Subject Area: Mathematics **Grade(s):** 9-12

Course Description:
This course 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 engages students in the creative aspects of the field by allowing them to development computational artifacts based on their interests. Students will also develop effective communication and collaborations skills by working individually and collaboratively to solve problems, and will discuss and write about the impacts these solutions could have on their community, society, and the world.

https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-course-and-exam-description.pdf

Time/Credit for the Course: Full Year / 1 credit

Curriculum Writing Committee: Jessica Hubal

Curriculum Map

1. **Marking Period One: Overview based on 45 days:**
   Introducing the Internet and digital information (encoding and compressing complex information).

   Students will explore the Internet. They will understand that the systems built on the Internet have had a profound impact on society. The principles of systems and networks that helped enable the Internet are also critical in the implementation of computational solutions. Students in this unit gain insight into how the Internet operates, how to study characteristics of the Internet and systems build on it, and how to analyze important concerns such as cybersecurity. Students will also explore the ways that digital information is encoded, represented and manipulated.

   Goals: Understanding of:
   - The internet is a network of autonomous systems.
   - Characteristics of the Internet influence the systems built on it.
   - Cybersecurity is an important concern for the Internet.
   - A variety of abstractions built on binary sequences can be used to represent all digital data.
   - Multiple levels of abstraction are used to write programs or create other computational artifacts.
   - Models and simulations use abstraction to generate new understanding and knowledge.

2. **Marking Period Two: Overview based on 45 days:**
   Continuation of digital information (manipulating and visualizing data), as well as introducing algorithms and programming.

   Students will further explore digital information by learning how to digitally manipulate data, visualize it, and identify patterns, trends and possible meanings are important practical skills that computer scientists do every day. They will understand where data comes from, having intuitions about what could be learned or extracted from it, and being able to use computational tools to manipulate data and communicate about it are the primary skills addressed in the unit. Students will also work with algorithms in many ways: they develop and express original algorithms, they implement algorithms in a language, and they analyze algorithms analytically and empirically. Students will learn fundamental concepts of programming that can be applied across a variety of projects and languages.

Goals: Understanding of:

- Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages.
- Algorithms can solve many, but not all, computational problems.
- Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society).
- Writing programs to execute algorithms.
- Programming is facilitated by appropriate abstractions.
- Programs are developed, maintained, and used by people for different purposes.
- Programming uses mathematical and logical concepts.

3. **Marking Period Three: Overview based on 45 days**
   Introducing implications of big data and focusing on user privacy, as well as concentrating on the implementation of innovative technologies.

   Students focus on the data-rich world they live in. Students investigate many complex questions related to public policy, law, ethics and societal impact. Students will develop a well-rounded and balanced view about data in the world, including the positive and negative effects of it, and to understand the basics of how and why modern encryption works. Students will learn how to use the tools and techniques of computer science to create interesting and relevant artifacts with characteristics that are enhanced by computation.

   Goals: Understanding of:

   - Computing enhances communication, interaction, and cognition.
   - Computing enables innovation in nearly every field.
   - Computing has global effects – both beneficial and harmful – on people and society.
   - Computing innovations influence and are influenced by the economic, social, and cultural contexts in which they are designed and used.
   - An investigative process is aided by effective organization and selection of resources. Appropriate technologies and tolls facilitate the accessing of information and enable the ability to evaluate the credibility of sources.

4. **Marking Period Four: Overview based on 45 days**
   Continuation of implementation of innovative technologies, as well as review for AP Computer Science Principles Multiple Choice Exam and prepare for AP Computer Science Principles Performance Tasks.

Students will continue to learn how to use the tools and techniques of computer science to create interesting and relevant artifacts with characteristics that are enhanced by computation. Students concentrate on ensuring their understanding and preparation for the AP Explore Performance Task. Students will also focus on practicing for the multiple choice portion of the AP Exam. The Post-AP overview allows students to explore and research other technical tools (either hardware or software) and create a final project.

Goals: Understanding of:

- People use computer programs to process information to gain insight and knowledge.
- Computing facilitates exploration and the discovery of connections in information.
- There are trade-offs when representing information as digital data.

Curriculum Plan

Unit 1: The Internet

Marking Period: 1st

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: The Internet pervades modern computing.

Essential Questions:

- What is the Internet? How is it built? How does it function?
- What aspects of the Internet's design and development have helped it scale and flourish?
- How is cybersecurity impacting the ever-increasing number of Internet users?

Concepts:

- The Internet connects devices and networks all over the world.
- The internet is built on evolving standards, including those for addresses and names.
- The domain name system (DNS) translates domain names to IP addresses.
- The internet and the systems built on it are hierarchical and redundant.
- Antivirus software and firewalls can help prevent unauthorized access to private data.
- An end-to-end architecture facilitates connecting new devices and networks on the Internet.
- Devices and networks that make up the Internet are connected and communicate using addresses and protocols.

Competencies:

- Explain the abstractions in the Internet and how the Internet functions.
- Explain characteristics of the Internet and the stems built on it.
- Identify existing cybersecurity concerns and potential options to address these issues with the Internet and the systems built on it.

Overview: In this unit, students will explore the Internet. They will understand that the systems built on the Internet have had a profound impact on society. The principles of systems and networks that helped enable the Internet are also critical in the implementation of computational solutions. Students in this unit gain insight into how the Internet operates, how to study characteristics of the Internet and systems build on it, and how to analyze important concerns such as cybersecurity.

Goals: Students will understand:

- The internet is a network of autonomous systems.
- Characteristics of the Internet influence the systems built on it.
- Cybersecurity is an important concern for the Internet.

Objectives:

- Define the internet. (DOK-1)
- Create and send binary messages. (DOK-4)
- Send encode and send numbers. (DOK-4)
- Develop encode and send text. (DOK-4)
- Investigate internet addresses, packets, & redundancy. (DOK-3)
- Create algorithms of the Internet, specifically routing. (DOK-4)
- Formulate protocols and abstraction using the internet. (DOK-3)

Core Activities and Corresponding Instructional Methods:

- Students will build personal innovations.
    - Examples:
        - Build a bit sending device
        - Send binary messages with the internet simulator
        - Encode and send number systems (circles, triangles, squares), binary numbers, formatted text
- Students will explore the internet by completing various activities on internet addressing, routers, redundancy, and reliability.
- Students invent a protocol that may use ASCII to encode formatted text such as fonts, colors, sizes, etc.

- Students will complete a performance task that allows them to research and prepare a discussion about social issues related to the Internet.
    - Example: The Internet and Society

❖ Direct instruction and practice on internet components.
❖ Guided practice: Students will development innovations that use all components.
❖ Group projects and presentations to enhance 21st century skills.
❖ Classroom discussion and guided practice on creating assignments/projects.
❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
    - Algebra 1 Final Grade
    - Computer Literacy / Video Game Design Final Grade
    - Programming 1 / Programming 2 / AP Computer Science A Final Grade

- **Formative:**
    - Diagnostic assessment and questioning
    - Quizzes
    - Practice worksheets and activity guides
        - Multiple choice or matching questions related to questions on the chapter summative assessment
        - Free-response text fields where students may input their answer
    - Graded worksheets
- **Summative:**
    - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
    - Practice Performance Task Assessments
    - Project Rubrics
        - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapters 1-2 in *Blown to Bits*
- Explore the Internet Access unit within the Computer Concepts 2018 textbook

Correctives:

- Re-teaching and practice programs available with textbook

Unit 2: Digital Information

Marking Period: 1st/2nd

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: Abstraction reduces information and detail to facilitate focus on relevant concepts.

Essential Questions:

- How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer?
- How does abstraction help us in writing programs, creating computational artifacts and solving problems?
- How can computational models and simulations help generate new understanding and knowledge?

Concepts:

- Digital data is represented by abstractions at different levels.
- The process of developing on abstraction involved removing detail and generalizing functionality.
- In an abstraction hierarchy, higher levels of abstraction would be placed toward the top and lower-level abstractions toward the bottom.
- Models and simulations re simplified representation of more complex objects and phenomena.

Competencies:

- Describe the variety of abstractions used to represent data.
- Explain how binary sequences are used to represent digital data.
- Develop an abstraction when writing a program or creating other computational artifacts.
- Use multiple levels of abstraction to write programs.

- Identify multiple levels of abstractions that are used when writing programs.
- Use models and simulations to represent phenomena.
- Use models and simulations to formulate, refine, and test hypotheses.

Big Idea #2: Communication includes written and oral descriptions supported by graphs, visualizations, and computational analysis.

Essential Questions:

- What different types of communications exist within the world?
- What creates effective communication?
- How can different forms of communication impact the purpose of a computational artifact?

Concepts:

- Describe the impact of technology and computation.
- Justify the design and appropriateness of computational choices.
- Analyze and describe both computational artifacts.

Competencies:

- 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.

Overview: This unit further explores the ways that digital information is encoded, represented and manipulated. Being able to digitally manipulate data, visualize it, and identify patterns, trends and possible meanings are important practical skills that computer scientists do every day. Understanding where data comes from, having intuitions about what could be learned or extracted from it, and being able to use computational tools to manipulate data and communicate about it are the primary skills addressed in the unit.

Goals: Students will understand:

- A variety of abstractions built on binary sequences can be used to represent all digital data.
- Multiple levels of abstraction are used to write programs or create other computational artifacts.
- Models and simulations use abstraction to generate new understanding and knowledge.

Objectives:

- Define text compression. (DOK-1)
- Encode black and white and color images. (DOK-3)
- Explore lossy compression and file formats. (DOK-3)
- Collect data. (DOK-2)

- Find trends with visualizations. (DOK-1)
- Construct data visualizations. (DOK-3)
- Clean data. (DOK-2)
- Analyze data. (DOK-4)

Core Activities and Corresponding Instructional Methods:

- Students will create a project/collage that requires text compression, encoding B&W images, encoding color images, lossy compression and file formats.
- Students will collect data by completing a project.
    - Example: Data Tracker Project
- Students will find trends in data and visualizations by completing an assignment.
    - Examples:
        - Check Your Assumptions assignment
        - Good and Bad Data Visualization assignment
- Students will enhance communication by incorporating visualization into projects.
    - Examples:
        - Discover a Data Story project
        - Creating Summary Tables assignment
        - Cleaning Data assignment
- Students will complete a performance task that allows them to research and prepare discussion on data and data visualizations.
    - Example:
        - Encode an Experience Performance Task
        - Tell a Data Story Performance Task

❖ Direct instruction and practice on encoding and compressing components, as well as manipulating and visualizing data.
❖ Guided practice: Students will development innovations that use all components.
❖ Group projects and presentations to enhance 21st century skills.
❖ Classroom discussion and guided practice on creating assignments/projects.
❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
    - Algebra 1 Final Grade
    - Computer Literacy / Video Game Design Final Grade
    - Programming 1 / Programming 2 / AP Computer Science A Final Grade

- **Formative:**
  - Diagnostic assessment and questioning
  - Quizzes
  - Practice worksheets and activity guides
    - Multiple choice or matching questions related to questions on the chapter summative assessment
    - Free-response text fields where students may input their answer
  - Graded worksheets

- **Summative:**
  - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
  - Practice Performance Task Assessments
  - Project Rubrics
    - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapters 3-4 in *Blown to Bits*
- Explore the Digital Devices and Digital Security units within the Computer Concepts 2018 textbook

Correctives:

- Re-teaching and practice programs available with textbook

Unit 3: Algorithms and Programming

Marking Period: 2nd

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: Algorithms are used to develop and express solutions to computational problems.

Essential Questions:

- How are algorithms implemented and executed on computers and computational devices?
- Why are some languages better than others when used to implement algorithms?
- What kinds of problems are easy, what kinds are difficult, and what kinds are impossible to solve algorithmically?
- How are algorithms evaluated?

Concepts:

- Sequencing, selection, and iteration are building blocks of algorithms.
- Sequencing is the application of each step of an algorithm in the order in which the statements are given.
- Algorithms can be combined to make new algorithms.
- Different algorithms can be developed to solve the same problem
- Languages for algorithms include natural language, pseudocode, and visual and textual programming languages.
- Reasonable time means that the number of steps the algorithm takes is less than or equal to a polynomial function (constant, linear, square, cube, etc.) of the size of the input.

Competencies:

- Develop an algorithm for implementation in a program.
- Express an algorithm in a language.
- Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time.
- Explain the difference between solvable and unsolvable problems in computer science.
- Explain the existence of undecidable problems in computer science.
- Evaluate algorithms analytically and empirically for efficiency, correctness, and clarity.

Big Idea #2: Programming enables problem solving, human expression, and creation of knowledge.

Essential Questions:

- How are programs developed to help people, organization, or society solve problems?
- How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?
- How do computer programs implement algorithms?
- How does abstraction make the development of computer programs possible?
- How do people develop and test computer programs?
- Which mathematical and logical concepts are fundamental to computer programming?

Concepts:

- Programs are developed and used in a variety of ways by a wide range of people depending on the goals of the programmer.
- 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.
- An iterative process of program development helps in developing a correct program to solve problems.
- Documentation about program components, such as code segments and procedures, helps in developing and maintain problems.

Competencies:

- Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge.
- Develop a correct program to solve problems.
- Collaborate to develop a program.
- Explain how programs implement algorithms.
- Use abstraction to manage complexity in programs.
- Evaluate the correctness of a program.
- Employ appropriate mathematical and logical concepts in programming.

Overview: In this unit, students will work with algorithms in many ways: they develop and express original algorithms, they implement algorithms in a language, and they analyze algorithms analytically and empirically. This unit also acquaints students with fundamental concepts of programming that can be applied across a variety of projects and languages. As students learn language specifics for a given programming language, they create programs, translating human intention into computational artifacts.

Goals: Students will understand:

- Algorithms are precise sequences of instructions for processes that can be executed by a computer and are implemented using programming languages.
- Algorithms can solve many, but not all, computational problems.
- Programs can be developed for creative expression, to satisfy personal curiosity, to create new knowledge, or to solve problems (to help people, organizations, or society).
- People write programs to execute algorithms.
- Programming is facilitated by appropriate abstractions.
- Programs are developed, maintained, and used by people for different purposes.
- Programming uses mathematical and logical concepts.

Objectives:

- Define programming. (DOK-1)
- Understand the need for programming languages. (DOK-1)
- Understand the need for algorithms. (DOK-1)
- Develop and formulate algorithms. (DOK-3)
- Analyze algorithms. (DOK-4)
- Design procedural code using top-down design. (DOK-4)
- Interpret and document code segments. (DOK-2)
- Create functions with parameters, looping structures, and random numbers. (DOK-4)

Core Activities and Corresponding Instructional Methods:

- Student will complete assignments that will allow them to analyze and construct creative algorithms.
- Students will construct a program that uses simple commands.
- Students will create a program that incorporates functions.
- Students will complete assignments that investigate APIs and function parameters.
- Students will construct a program that uses looping and random numbers.
- Students will complete a performance task that allows them to research and prepare a program.
    - Example: Design a Digital Scene

- ❖ Direct instruction and practice on constructing code.
- ❖ Guided practice: Students will development innovations that use all coding components.
- ❖ Group projects and presentations to enhance 21st century skills.
- ❖ Classroom discussion and guided practice on creating assignments/projects.
- ❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
  - Algebra 1 Final Grade
  - Computer Literacy / Video Game Design Final Grade
  - Programming 1 / Programming 2 / AP Computer Science A Final Grade
- **Formative:**
  - Diagnostic assessment and questioning
  - Quizzes
  - Practice worksheets and activity guides
    - Multiple choice or matching questions related to questions on the chapter summative assessment
    - Free-response text fields where students may input their answer
  - Graded worksheets
- **Summative:**
  - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
  - Practice Performance Task Assessments
  - Project Rubrics
    - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapter 5 in *Blown to Bits*
- Explore the Programming unit within the Computer Concepts 2018 textbook

Correctives:

- Re-teaching and practice programs available with textbook

Unit 4: Big Data and Privacy

Marking Period: 3rd

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: Computing has a global impact.

Essential Questions:

- How does computing enhance human communication, iteration, and cognition?
- How does computing enable innovation?
- What are some potential beneficial and harmful effects of computing?
- How do economic, social, and cultural contexts influence innovation and the use of computing?

Concepts:

- Email, SMS, and chat have fostered new ways to communicate and collaborate.
- Public data provides widespread access to information facilitates the identification of problems, development of solutions, and dissemination of results.
- Distributed solutions must scale to solve some problems.
- Technology enables the collection, use, and exploitation of information about, by, and for individuals, groups, and institutions.

Competencies:

- Explain how computing innovations affect communication, integration, and cognition.
- Explain how people participate in a problem-solving process that scales.
- Explain how computing has impacted innovations in other fields.
- Analyze the beneficial and harmful effects of computing.
- Explain the connections between computing and real-world contexts, including economic, social, and cultural contexts.

- Access, manage, and attribute information using effective strategies.
- Evaluate online and print sources for appropriateness and credibility.

Overview: This unit introduces students to the data-rich world they live in. Students investigate many complex questions related to public policy, law, ethics and societal impact. The goals of this unit are to develop a well-rounded and balanced view about data in the world, including the positive and negative effects of it, and to understand the basics of how and why modern encryption works.

Goals: Students will understand:

- Computing enhances communication, interaction, and cognition.
- Computing enables innovation in nearly every field.
- Computing has global effects – both beneficial and harmful – on people and society.
- Computing innovations influence and are influenced by the economic, social, and cultural contexts in which they are designed and used.
- An investigative process is aided by effective organization and selection of resources. Appropriate technologies and tolls facilitate the accessing of information and enable the ability to evaluate the credibility of sources.

Objectives:

- Define Big Data. (DOK-1)
- Recognize data in the real world. (DOK-1)
- Identify people with data. (DOK-1)
- Investigate the cost of "free" in respect to technology. (DOK-3)
- Analyze the security/privacy dilemma. (DOK-4)
- Identify the need for encryption. (DOK-1)
- Connect the concepts of security and symmetric encryption. (DOK-4)
- Relate asymmetric encryption and public keys. (DOK-4)

Core Activities and Corresponding Instructional Methods:

- Students will complete activities to emphasize Big Data and how it connects to real world.
- Students will examine data innovations through a project.
    - Example: Rapid Research – Data Innovations
- Students will complete an assignment that allows them to identify people with data.
- Students will observe and investigate encryption through a project.
    - Example: Cracking the Code
- Students will complete a performance task that allows them to research and prepare a discussion on big data.
    - Example: The Big Data Dilemma

- ❖ Direct instruction and practice on Big Data and encryption.
- ❖ Guided practice: Students will development innovations that use all components.
- ❖ Group projects and presentations to enhance 21st century skills.
- ❖ Classroom discussion and guided practice on creating assignments/projects.
- ❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
  - Algebra 1 Final Grade
  - Computer Literacy / Video Game Design Final Grade
  - Programming 1 / Programming 2 / AP Computer Science A Final Grade
- **Formative:**
  - Diagnostic assessment and questioning
  - Quizzes
  - Practice worksheets and activity guides
    - Multiple choice or matching questions related to questions on the chapter summative assessment
    - Free-response text fields where students may input their answer
  - Graded worksheets
- **Summative:**
  - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
  - Practice Performance Task Assessments
  - Project Rubrics
    - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapter 6 in *Blown to Bits*
- Explore the Databases unit within the Computer Concepts 2018 textbook

Correctives:

- Re-teaching and practice programs available with textbook

Unit 5: Innovation Technologies

Marking Period: 3rd/ 4th

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: Computing is a creative activity.

Essential Questions:

- How can a creative development process affect the creation of computation artifacts?
- How can computing and the use of computational tools foster creative expression?
- How can computing extend traditional forms of human expression and experience?

Concepts:

- 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.
- Creating computational artifacts employs an iterative and often exploratory process translates ideas into tangible form.
- A computational artifact is something created by a human using a computer and can be, but is not limited to, a program, an image, an audio, a video, a presentation, or a Web page file.
- Creating computational artifacts requires understanding of and use of software tools and services.

Competencies:

- Apply a creative development process when creating computational artifacts.
- Create a computational artifact for creative expression.
- Create a computational artifact using computing tools and techniques to solve a problem.
- Create a new computational artifact by combing or modifying existing artifacts.
- Collaborate in the creation of computational artifacts.
- Analyze the correctness, usability, functionality, and suitability of computational artifacts.
- Use computing tools and techniques for creative expression.

Overview: This unit continues the introduction of foundational concepts of computer programming, which unlocks the ability to make rich, interactive apps. It emphasizes the creative aspects of computing. Students in this unit will learn how to use the tools and techniques of computer science to create interesting and relevant artifacts with characteristics that are enhanced by computation.

Goals: Students will understand:

- Creative development can be an essential process for creating computational artifacts.
- Computing enables people to use creative development processes to create computational artifacts for creative expression or to solve a problem.
- Computing can extend traditional forms of human expression and experience.
- People use computer programs to process information to gain insight and knowledge.
- Computing facilitates exploration and the discovery of connections in information.
- There are trade-offs when representing information as digital data.
- Communication gives rise to enhanced insights and knowledge.
- Communication includes written and oral descriptions supported by graphs, visualization, and computational analysis.

Objectives:

- Explore event driven programming and applications. (DOK-3)
- Incorporate variables, strings, conditionals, and Boolean logic. (DOK-2)
- Apply the concept of program flow and logic conditionals. (DOK-4)
- Identify digital assistants. (DOK-1)
- Design algorithms that use looping structures and arrays. (DOK-4)
- Build apps. (DOK-4)

Core Activities and Corresponding Instructional Methods:

- Students will complete activities that promote controlling memory with variables, using variables in apps, and user input/strings.
- Students will complete assignments to enhance understanding of program flow and logic conditionals.
- Students will code programs that incorporate loops and arrays.
- Students will program their own application design.
- Students will complete a performance task that allows them to research and prepare an application.
    - Example: Improve Your App

❖ Direct instruction and practice on processes of writing apps.
❖ Guided practice: Students will development innovations that use all app components.
❖ Group projects and presentations to enhance 21st century skills.
❖ Classroom discussion and guided practice on creating assignments/projects.
❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
    - Algebra 1 Final Grade
    - Computer Literacy / Video Game Design Final Grade
    - Programming 1 / Programming 2 / AP Computer Science A Final Grade
- **Formative:**
    - Diagnostic assessment and questioning
    - Quizzes
    - Practice worksheets and activity guides
        - Multiple choice or matching questions related to questions on the chapter summative assessment
        - Free-response text fields where students may input their answer
    - Graded worksheets
- **Summative:**
    - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
    - Practice Performance Task Assessments
    - Project Rubrics
        - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapter 7 in *Blown to Bits*
- Explore the Programming unit within the Computer Concepts 2018 textbook

Correctives:

- Re-teaching and practice programs available with textbook

Unit 6: Performance Tasks and Post-AP Final Project

Marking Period: 4th

Standard(s):

- Science and Technology Secondary Standards
  - 3.4.10.A1, 3.4.10.A2, 3.4.10.A3, 3.4.10.B1, 3.4.10.B2, 3.4.10.B3, 3.4.10.B4, 3.4.10.C1, 3.4.10.C2, 3.4.10.3.C3, 3.4.10.D1, 3.4.10.D2, 3.4.10.D3, 3.4.10.E1, 3.4.10.E2, 3.4.10.E3, 3.4.10.E4, 3.4.10.E5, 3.4.10.E6 http://static.pdesas.org/content/documents/Academic_Standards_for_Science_and_Technology_and_Engineering_Education_(Secondary).pdf

- Business, Computer and Information Technology Standards
  - 15.3.12.E, 15.3.12.F, 15.3.12.G, 15.3.12.H, 15.3.12.I, 15.3.12.M, 15.3.12.P, 15.3.12.S, 15.3.12.W, 15.4.12.A, 15.4.12.B, 15.4.12.C, 15.4.12.D, 15.4.12.E, 15.4.12.F, 15.4.12.G, 15.4.12.H, 15.4.12.I, 15.4.12.J, 15.4.12.K, 15.4.12.L, 15.4.12.M http://static.pdesas.org/content/documents/BCIT_standards.pdf

Big Idea #1: Data and information facilitate the creation of knowledge.

Essential Questions:

- How can computation be employed to help people process data and information to gain insight and knowledge?
- How can computation be employed to facilitate exploration and discovery when working with data?
- What considerations and trade-offs arise in the computational manipulation of data?
- What opportunities do large data sets provide for solving problems and creating knowledge.

Concepts:

- Computers are used in an iterative and interactive way when processing digital information to gain insight and knowledge.
- Digital information can be filtered and cleaned by using computers to process information.
- Combining data sources, clustering data, and data classification are part of the process of using computers to process information.

Competencies:

- Find patterns and test hypotheses about digitally processed information to gain insight and knowledge.
- Collaborate when processing information to gain insight and knowledge.
- Explain the insight and knowledge gained from digitally processed data by using appropriate visualization, notations, and precise language.
- Extract information from data to discover and explain connections or trends.

- Determine how large data sets impact the use of computational processes to discover information and knowledge.
- Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information.

Overview: This unit concentrates on ensuring that students understand, prepare for, and do the AP Explore Performance Task. Students will also focus on practicing for the multiple choice portion of the AP Exam. The Post-AP overview allows students to explore and research other technical tools (either hardware or software) and create a final project.

Goals: Students will understand:

- People use computer programs to process information to gain insight and knowledge.
- Computing facilitates exploration and the discovery of connections in information.
- There are trade-offs when representing information as digital data.

Objectives:

- Recall requirements for the AP Computer Science Principles exam. (DOK-1)
- Analyze strategies for managing time during the AP Computer Science Principles Exam. (DOK-4)
- Apply concepts learned throughout the course to complete Performance Tasks. (DOK-4).
- Experience the administration of *Create* Performance Task (12 hours' requirement). (DOK-4)
- Experience the administration of *Explore* Performance Task (8 hours' requirement). (DOK-4)
- Apply all concepts from the course to create a well-developed, creative, personal innovation. (DOK-4)
- Construct a flowchart. (DOK-2)
- Construct a presentation based on the process involved for developing the personal innovation. (DOK-3)

Core Activities and Corresponding Instructional Methods:

- Students will complete activities that help prepare them for the Multiple-Choice section of the AP Computer Science Principles Exam.
- Students will complete activities that help prepare them for the Performance Tasks section of the AP Computer Science Principles Exam.
- Students will complete the Create and Explore Performance Tasks.
- After the AP test, students will create a well-developed, creative, personal innovation.

- Students will create a flowchart that represents their personal innovation.
- Students will create a presentation that showcases the processes involved while creating their personal innovation.

❖ Direct instruction and practice on multiple-choice and performance tasks.
❖ Guided practice: Students will development innovations that use all components.
❖ Group projects and presentations to enhance 21st century skills.
❖ Classroom discussion and guided practice on creating assignments/projects.
❖ Integration of single platform called Code Studio.

Assessments:

- **Diagnostic:**
  - Algebra 1 Final Grade
  - Computer Literacy / Video Game Design Final Grade
  - Programming 1 / Programming 2 / AP Computer Science A Final Grade
- **Formative:**
  - Diagnostic assessment and questioning
  - Quizzes
  - Practice worksheets and activity guides
    - Multiple choice or matching questions related to questions on the chapter summative assessment
    - Free-response text fields where students may input their answer
  - Graded worksheets
- **Summative:**
  - Fixed Responses Assessments (multiple choice, matching, choose two, short answer, etc.)
  - Practice Performance Task Assessments
  - Project Rubrics
    - Written and project work, practice performance task, programming projects, student presentations

Extensions:

- AP Computer Science Principles Practice Multiple Choice Exam Practice
- Read Chapter 8 in *Blown to Bits*
- Review AP Computer Science Principles test prep books

Correctives:

- Re-teaching and practice programs available with textbook

Materials and Resources (All Units):

- *New Perspective: Computer Concepts 2018* Textbook
- College Board AP Computer Science Principles Class Resources (https://apcentral.collegeboard.org/courses/ap-computer-science-principles/classroom-resources)
- Code: Computer Science Principles Course (https://studio.code.org/courses/csp)
- *Blown to* Bits Textbook (Free download: http://www.bitsbook.com/)

Primary Textbook(s) Used for this Course of Instruction

Name of Textbook: *New Perspective: Computer Concepts 2018*

Textbook ISBN #: 978-1-305-95149-5

Textbook Publisher &Year of Publication: Cengage 2018

Curriculum Textbook is utilized in (title of course): AP Computer Science Principles

Checklist to Complete and Submit:

(Scan and email)

_____ Copy of the curriculum using the template entitled "Planned Instruction," available on the district website.

_____ The primary textbook form(s).

_____ The appropriate payment form, in compliance with the maximum curriculum writing hours noted on the first page of this document.

Each principal and/or department chair has a schedule of First and Second Readers/Reviewers. Each Reader/Reviewer must sign & date below.

First Reader/Reviewer Printed Name_____

First Reader/Reviewer Signature_____ Date_____

Second Reader/Reviewer Printed Name_____

Second Reader/Reviewer Signature _____ Date_____