

# Wallenpaupack Area School District Planned Course Curriculum Guide

## Department

Business, Computers, and Information Technology (BCIT)/CTE

## Name of Course

Computer Programming

### Course Description:

This course is an introductory course in object-oriented computer languages and general computer science. This course is intended to provide familiarity with general-purpose programming by introducing fundamental programming concepts and programming languages. The process of defining a problem, breaking it down into a series of smaller problems and finally writing the code to solve it is followed. This course is a great introduction to both fundamental programming concepts and programming languages in addition to providing a solid background in general computer science understanding. Logical thinking skills and proper programming techniques are emphasized.

**Initial Creation Date (if applicable) and Revision Dates:** October 2024

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 1:</b> Beginning in Computer Science	<b>TIMEFRAME:</b> 2 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-IC-24 – Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.</li> <li>• 3A-CS-02 – Compare levels of abstraction and interactions between application software, system software, and hardware layers.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Define computer science and coding.</li> <li>• Define hardware and software.</li> <li>• Define and utilize variables.</li> <li>• Define and distinguish between analog and digital data.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> Computer Science, Python, natural language, programming language, hardware, software, input, output</p>

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 2:</b> Number Calculations and Data	<b>TIMEFRAME:</b> 1.5 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-DA-09 – Translate between different bit representations of real-world phenomena, such as characters, numbers, and images.</li> <li>• 3A-DA-10 – Evaluate the tradeoffs in how data elements are organized and where data is stored.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Briefly describe the history of the computer.</li> <li>• Use Python to do numerical calculations.</li> <li>• Define and utilize modular division.</li> <li>• Define and utilize import, functions, and modules.</li> <li>• Generate random numbers with random functions.</li> <li>• Define and utilize maximum and minimum.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> Operator, modular division, function, import, module, random, maximum, minimum, terabyte</p>

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 3:</b> Functions	<b>TIMEFRAME:</b> 1.5 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</li> <li>• 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Explain the purpose of functions</li> <li>• Define and understand abstraction</li> <li>• Understand the use of parameters</li> <li>• Define and code if, and if-else statements.</li> <li>• Use logical operators to create Boolean expressions</li> <li>• Define, analyze, utilize, and list the rules and standards of a good algorithm.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> function, parameter, if statement, relational operators, syntax, Boolean Expression, nested ifs, algorithm</p>

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 4:</b> Repetition and Loops	<b>TIMEFRAME:</b> 2 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</li> <li>• 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</li> <li>• 3A-AP-16 – Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Define and code while loops and loop control variables.</li> <li>• Define and code the count and sum variables.</li> <li>• End loops using count variables or user input, and state when to use them.</li> <li>• Code two types of while loops.</li> <li>• Define and utilize the range function.</li> <li>• Define and code for loops.</li> <li>• Code for loops with control and sum variables.</li> <li>• Define and utilize algorithms and code tracing.</li> <li>• Define computer models and simulations and explain why these are used.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> count variable, loop, sum variable, while loop, for loop, control variable, function, parameter, range, initialization, algorithm, tracing code, modeling, simulation</p>

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 5:</b> Programming in EarSketch	<b>TIMEFRAME:</b> 2 Weeks

**PA COMMON CORE/NATIONAL STANDARDS:**

- 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

**UNIT OBJECTIVES (SWBATS):**

- Identify and use the basic parts of the EarSketch platform.
- Identify the required components of an EarSketch script.
- Define rhythm, tempo, and beats.
- Use basic functions in EarSketch to write a short piece of music.
- Change the qualities of a sound in your EarSketch code using the setEffect function.
- Use the setEffect function in EarSketch to vary the length of time that an effect occurs for.
- Define tempo and pitch.
- Use setTempo to alter the speed of a music sample in EarSketch.
- Define and explain fair use, as it relates to music creation.
- Change the duration, complexity, or quality of your music in EarSketch using a for or while loop.
- Use concatenation and other operations to change strings in your musical code.

**INSTRUCTIONAL STRATEGIES/ACTIVITIES:**

- Online Project STEM Course and LMS
- Class Discussions
- Direct Instruction

**ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):**

- Lesson Practices
- Quizzes
- Unit Project
- Unit Test

**DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):**

- Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.
- Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.

**RESOURCES (Technology Based Resources, Text Resources, etc.):**

- Project STEM Website and LMS
- Laptop/Desktop/iPad
- EarSketch

**KEY VOCABULARY:** rhythm, tempo, beats, pitch, fair use, concatenation, copyright, sampling, string, substrings, repetition, contrast

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 6:</b> Graphics	<b>TIMEFRAME:</b> 1.5 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</li> <li>• 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</li> <li>• 3A-AP-18 – Create artifacts by using procedures with a program, combinations of data and procedures, or independent but interrelated programs.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Use the random function to generate random colors.</li> <li>• Define and utilize RGB and hexadecimal format to create colors.</li> <li>• Define and code the simplegui.create_frame() function.</li> <li>• Code for loops and range functions within the simplegui package.</li> <li>• Use pairs of (x,y) coordinates, frames, and the def function in code.</li> <li>• Create drawings using commands and functions.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> RGB, hexadecimal, console, coordinate plane, frame</p>

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 7:</b> Functions	<b>TIMEFRAME:</b> 1.5 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</li> <li>• 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</li> <li>• 3A-AP-15 – Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program, and explain the benefits and drawbacks of choices made.</li> <li>• 3A-AP-21 – Evaluate and refine computational artifacts to make them more usable and accessible.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Explain the purpose of a function in programming.</li> <li>• Define functions and call them in a program.</li> <li>• Define a parameter and argument.</li> <li>• Define and create a function that uses optional parameters.</li> <li>• Perform operations on strings using functions.</li> <li>• Code functions that return values and use the returned values in your code.</li> <li>• Understand how a more complex program can use several different functions together.</li> <li>• Define the term local variable.</li> <li>• Understand when a program can access a local variable.</li> <li>• Trace through existing code to discover what it does.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>
<p><b>KEY VOCABULARY:</b> subprogram, def, main, parameter, string, isdigit, lower, upper, replace, str, return, local variable, global variable, tracing code</p>



<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 8:</b> Lists	<b>TIMEFRAME:</b> 2 Weeks

**PA COMMON CORE/NATIONAL STANDARDS:**

- 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.
- 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
- 3A-AP-14 – Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.

**UNIT OBJECTIVES (SWBATS):**

- Define lists and explain their usefulness in programming
- Declare and add data to lists in Python.
- Define the terms element, index, and initializer list as they relate to lists.
- Work out which index relates to which item in a list.
- Write code to access elements at given indices in a list.
- Use a loop to add data and process existing data in a list.
- Explain the function of a variety of methods for lists in Python, and write code using these methods.
- Code functions that use lists as parameters.
- Explain the difference between how Python handles parameters that are primitive data types and parameters that are lists.
- Use string methods to process the individual strings in a list.
- Define and utilize parallel lists.
- Describe several different algorithms that are used to sort data into an order.
- Code a simple search function and search for specific values within data.
- Code a simple sort function that takes a list as an input and returns the list with the values ordered alphabetically or numerically.

**INSTRUCTIONAL STRATEGIES/ACTIVITIES:**

- Online Project STEM Course and LMS
- Class Discussions
- Direct Instruction

**ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):**

- Lesson Practices
- Quizzes
- Unit Project
- Unit Test

**DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):**

- Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.
- Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.

**RESOURCES (Technology Based Resources, Text Resources, etc.):**

- Project STEM Website and LMS
- Laptop/Desktop/iPad

**KEY VOCABULARY:** list, declare, append, element, index, initializer list, extend, insert, pop, remove, sort, parameter, dot notation, parallel lists, sort, swap, array.

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 9:</b> 2D Lists	<b>TIMEFRAME:</b> 1 Week

**PA COMMON CORE/NATIONAL STANDARDS:**

- 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.
- 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
- 3A-AP-14 – Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.

**UNIT OBJECTIVES (SWBATS):**

- Define a 2D list.
- State the order of the row and column numbers in a 2D list address.
- Declare a 2D list, add data to a 2D list, and access data in a 2D list.
- Add data to, and process data in, a 2D list by using nested for loops.
- Code functions which build 2D lists from given parameters, and process the data stored in 2D lists.
- Explain how pixel data for an image may be stored in a 2D list.
- Write code that manipulates a simple image stored in a 2D list.
- Trace code that uses a 2D list to create the effect of animation on the screen.

**INSTRUCTIONAL STRATEGIES/ACTIVITIES:**

- Online Project STEM Course and LMS
- Class Discussions
- Direct Instruction

**ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):**

- Lesson Practices
- Quizzes
- Unit Project
- Unit Test

**DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):**

- Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.
- Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.

**RESOURCES (Technology Based Resources, Text Resources, etc.):**

- Project STEM Website and LMS
- Laptop/Desktop/iPad

**KEY VOCABULARY:** Two-dimensional list, column, row, global variable, row-major**Wallenpaupack Area School District Curriculum****COURSE:** Computer Programming**GRADE/S:** 10-12**UNIT 10:** Programming in EarSketch**TIMEFRAME:** 1 Week**PA COMMON CORE/NATIONAL STANDARDS:**

- 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

**UNIT OBJECTIVES (SWBATS):**

- Locate errors in your code and debug scripts by using the print function and observing output.
- Evaluate your own code and assess its completeness and correctness.
- Explain the use of algorithms and Booleans in programming.
- Use algorithms and Booleans in your EarSketch script.
- Explain how list operations are used in EarSketch.
- Apply random functions to scripts in EarSketch.

**INSTRUCTIONAL STRATEGIES/ACTIVITIES:**

- Online Project STEM Course and LMS
- Class Discussions
- Direct Instruction

**ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):**

- Lesson Practices
- Quizzes
- Unit Project
- Unit Test

**DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):**

- Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.
- Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.

**RESOURCES (Technology Based Resources, Text Resources, etc.):**

- Project STEM Website and LMS
- Laptop/Desktop/iPad
- EarSketch

**KEY VOCABULARY:** debug, algorithm, Boolean

<b>Wallenpaupack Area School District Curriculum</b>	
<b>COURSE:</b> Computer Programming	<b>GRADE/S:</b> 10-12
<b>UNIT 11:</b> Internet	<b>TIMEFRAME:</b> 1.5 Weeks

<p><b>PA COMMON CORE/NATIONAL STANDARDS:</b></p> <ul style="list-style-type: none"> <li>• 3A-AP-13 – Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.</li> <li>• 3A-AP-17 – Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.</li> <li>• 3A-NI-04 – Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.</li> <li>• 3A-NI-06 – Recommend security measures to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts.</li> </ul>
<p><b>UNIT OBJECTIVES (SWBATS):</b></p> <ul style="list-style-type: none"> <li>• Explain how bits of data are sent over the internet using copper wires, fiber optic cable, and radio waves.</li> <li>• Explain how IP addresses and the Domain Name System allow data to be “addressed” to a device on the internet.</li> <li>• Explain how data is split into packets, routed to its intended location, and reassembled and verified when received.</li> <li>• Write a simple web page using HTML.</li> <li>• Be able to explain how HTTP is used to send HTML documents across the internet.</li> <li>• Format content on a webpage using HTML tags and stylesheets.</li> <li>• Create web pages using HTML that incorporate images.</li> <li>• Explain several cybersecurity threats and how to protect yourself from them.</li> <li>• Be able to state how encryption is used to transfer data securely on the internet.</li> <li>• Explain the principle of net neutrality.</li> <li>• State arguments for and against net neutrality.</li> </ul>
<p><b>INSTRUCTIONAL STRATEGIES/ACTIVITIES:</b></p> <ul style="list-style-type: none"> <li>• Online Project STEM Course and LMS</li> <li>• Class Discussions</li> <li>• Direct Instruction</li> </ul>
<p><b>ASSESSMENTS (Diagnostic/Benchmark/Formative/Summative):</b></p> <ul style="list-style-type: none"> <li>• Lesson Practices</li> <li>• Quizzes</li> <li>• Unit Project</li> <li>• Unit Test</li> </ul>
<p><b>DIFFERENTIATED INSTRUCTION (Acceleration/Enrichment):</b></p> <ul style="list-style-type: none"> <li>• Students work at their own pace throughout this course with teachers adding enrichment where needed based on individual strengths. Accelerated students have the opportunity to finish the course and complete independent projects.</li> <li>• Extension projects are incorporated into each unit and may be used at teacher discretion with accelerated students based on their individual interests.</li> </ul>
<p><b>RESOURCES (Technology Based Resources, Text Resources, etc.):</b></p> <ul style="list-style-type: none"> <li>• Project STEM Website and LMS</li> <li>• Laptop/Desktop/iPad</li> </ul>

**KEY VOCABULARY:** internet, network, bandwidth, bitrate, latency, protocol, Internet Service Provider, Internet Protocol, Domain Name System, packet, router, Transmission Control Protocol, HTML, attribute, CSS, Cybersecurity, virus, encryption, decryption, net neutrality