Unit 1: Computational Thinking

Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>1A</td>
<td>Unit Project</td>
<td>Password Generator Project</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
<tr>
<td>1B</td>
<td>Algorithmic Thinking</td>
<td>Problem Solving</td>
<td>EU 2.2, EU 4.1, EU 4.2, EU 7.2</td>
<td>LO 2.2.3, LO 4.1.1, LO 4.1.2, LO 4.2.2, LO 4.2.4, LO 7.2.1</td>
</tr>
<tr>
<td>1C</td>
<td>Algorithmic Thinking</td>
<td>Flow Patterns</td>
<td>EU 2.2, EU 4.1, EU 4.2, EU 7.2</td>
<td>LO 2.2.3, LO 4.1.1, LO 4.1.2, LO 4.2.2, LO 4.2.4, LO 7.2.1</td>
</tr>
<tr>
<td>1D</td>
<td>Algorithmic Thinking</td>
<td>Searching and Sorting</td>
<td>EU 2.2, EU 4.1, EU 4.2, EU 7.2</td>
<td>LO 2.2.3, LO 4.1.1, LO 4.1.2, LO 4.2.2, LO 4.2.4, LO 7.2.1</td>
</tr>
<tr>
<td>1E</td>
<td>Coding Skills</td>
<td>Encryption</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.2, LO 4.2.3, LO 6.3.1</td>
</tr>
<tr>
<td>1F</td>
<td>Big Picture</td>
<td>Cybersecurity</td>
<td>EU 6.3</td>
<td>LO 6.3.1</td>
</tr>
<tr>
<td>1G</td>
<td>Programming Languages</td>
<td>Grammar, Vocabulary, and Syntax</td>
<td>EU 2.2</td>
<td>LO 2.2.3</td>
</tr>
<tr>
<td>1H</td>
<td>Programming Languages</td>
<td>High-level vs. Low-level Languages</td>
<td>EU 2.2</td>
<td>LO 2.2.3</td>
</tr>
<tr>
<td>1I</td>
<td>Programming Languages</td>
<td>How a Computer Runs a Program</td>
<td>EU 2.2</td>
<td>LO 2.2.3</td>
</tr>
<tr>
<td>1J</td>
<td>Unit Project</td>
<td>Password Generator Project</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
<tr>
<td>1K</td>
<td>Coding Skills</td>
<td>Pseudocode</td>
<td>EU 2.2, EU 4.1</td>
<td>LO 2.2.3, LO 4.1.1, LO 4.1.2</td>
</tr>
<tr>
<td>1L</td>
<td>Solvability and Performance</td>
<td>Decidability</td>
<td>EU 4.1, EU 4.2, EU 5.2, EU 7.2</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.2, LO 4.2.3, LO 4.2.4, LO 5.2.1, LO 7.2.1</td>
</tr>
<tr>
<td>1M</td>
<td>Solvability and Performance</td>
<td>Algorithm Efficiency</td>
<td>EU 4.1, EU 4.2, EU 5.2, EU 7.2</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.2, LO 4.2.3, LO 4.2.4, LO 5.2.1, LO 7.2.1</td>
</tr>
<tr>
<td>1N</td>
<td>Solvability and Performance</td>
<td>Scalability of Solutions</td>
<td>EU 4.1, EU 4.2, EU 5.2, EU 7.2</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.2, LO 4.2.3, LO 4.2.4, LO 5.2.1, LO 7.2.1</td>
</tr>
<tr>
<td>1O</td>
<td>Unit Project</td>
<td>Password Generator Project</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
<tr>
<td>1P</td>
<td>Big Picture</td>
<td>Moore’s Law</td>
<td>EU 7.2</td>
<td>LO 7.2.1</td>
</tr>
<tr>
<td>1Q</td>
<td>Unit Project</td>
<td>Password Generator Project</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
<tr>
<td>1R</td>
<td>Unit Project</td>
<td>Password Generator Project</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
<tr>
<td>1S</td>
<td>Unit Exam</td>
<td>Unit 1 Exam</td>
<td>EU 2.2, EU 4.1, EU 4.2, EU 5.2, EU 6.3, EU 7.2</td>
<td>LO 2.2.3, LO 4.1.1, LO 4.1.2, LO 4.2.1, LO 4.2.2, LO 4.2.3, LO 4.2.4, LO 5.2.1, LO 6.3.1, LO 7.2.1</td>
</tr>
<tr>
<td>1T</td>
<td>Unit Project</td>
<td>Project Presentations</td>
<td>EU 4.1, EU 4.2, EU 6.3</td>
<td>LO 4.1.1, LO 4.2.1, LO 4.2.3, LO 4.2.4, LO 6.3.1</td>
</tr>
</tbody>
</table>

Unit Description

In order to successfully master the art of creating computational artifacts, it is important that students develop a clear understanding of the complex processes and structures that make up an algorithmic solution to a given problem. In addition, it is critical that they be able to formally
Unit 1: Computational Thinking

express those solutions clearly and unambiguously, such as what can be achieved through the use of pseudocode or a well-specified programming language. This unit focuses on introducing students to these concepts and helping them to develop the skills that they will rely on throughout the remainder of the course.

First, students will explore a number of techniques for analyzing common problems and visualizing their solutions. They will use these techniques to investigate a number of real-world applications, such as searching, sorting, and encryption. Next, students will examine how programmers utilize various levels of abstraction in the languages that they use to write programs and communicate their intentions in a form that can be executed by a computer. Finally, students will turn their attention to the question of whether various problems are solvable and investigate the factors that affect the efficiency of a solution to a given problem.

Unit Coverage

- Big Ideas: BI 2, BI 4, BI 5, BI 6, BI 7
- Enduring Understandings: EU 2.2, EU 4.1, EU 4.2, EU 5.2, EU 6.3, EU 7.2
- Learning Objectives: LO 2.2.3 [P], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 4.2.1 [P1], LO 4.2.2 [P1], LO 4.2.3 [P1], LO 4.2.4 [P4], LO 5.2.1 [P3], LO 6.3.1 [P1], LO 7.2.1 [P1]
- Computational Thinking Practices: P1, P2, P3, P4, P5

Unit Topics

- Algorithmic Thinking [EU 2.2, EU 4.1, EU 4.2, EU 7.2] [LO 2.2.3 [P3], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 4.2.2 [P1], LO 4.2.4 [P4], LO 7.2.1 [P1]]
  - Students will examine strategies for approaching large-scale problems.
  - Students will explore the benefits and applications of employing a top-down approach to problem solving.
  - Students will explore the benefits and applications of employing a bottom-up approach to problem solving.
  - Students will identify and examine a number of common features of algorithms, including sequencing, selection, and repetition.
  - Students will compare the differences between sequential search and binary search algorithms.
  - Students will collaboratively design and evaluate a sorting algorithm.
  - Students will compare the methods and relative efficiencies of different sorting algorithms.
- Programming Languages [EU 2.2] [LO 2.2.3 [P3]]
  - Students will examine the need for clarity and precision in communicating an algorithmic solution to a problem.
  - Students will examine the shortcomings and ambiguities of natural languages.
  - Students will identify the elements of clear communication, including well-specified grammar, vocabulary, and syntax.
  - Students will analyze the need for artificial programming languages.
Unit 1: Computational Thinking

- Students will compare high-level languages with low-level languages.
- Students will examine the process in which a program is written in a high-level language, compiled into a low-level language, loaded into memory, and then executed by a processor.
- Solvability and Performance [EU 4.1, EU 4.2, EU 5.2, EU 7.2] [LO 4.1.1 [P2], LO 4.2.1 [P1], LO 4.2.2 [P1], LO 4.2.3 [P1], LO 4.2.4 [P4], LO 5.2.1 [P3], LO 7.2.1 [P1]]
  - Students will examine the factors that affect the decidability of a problem.
  - Students will identify which problems can and cannot always be solved by an algorithm.
  - Students will examine methods of comparing equivalent algorithms for relative efficiency.
  - Students will evaluate the relative efficiency of equivalent algorithms.
  - Students will identify factors that allow solutions to scale efficiently.
- Coding Skills [EU 2.2, EU 4.1, EU 4.2, EU 6.3] [LO 2.2.3 [P3], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 4.2.2 [P1], LO 4.2.3 [P1], LO 6.3.1 [P1]]
  - Students will identify the needs and applications of cryptography in our digital world.
  - Students will analyze the differences between symmetric (single-key) encryption and asymmetric (public key) encryption.
  - Students will examine the mathematical foundation of cryptography.
  - Students will encode and decode messages using common cryptographic techniques.
- Big Picture [EU 7.2] [LO 7.2.1 [P1]]
  - Students will examine a number of common threats to cybersecurity, including distributed denial of service attacks (DDoS), phishing, viruses, and social engineering.
  - Students will identify the needs for robust cybersecurity.
  - Students will analyze the software, hardware, and human components of cybersecurity.
  - Students will analyze the function and effectiveness of common cybersecurity solutions, including antivirus software and firewalls.
  - Students will examine the implications of Moore's Law on the research and development of new and existing technologies.

Unit Project [CR1a] [CR1b] [CR1d] [CR2d] [CR2f]

- Password Generator Project
  - Students will collaborate in pairs to design an algorithm for generating a custom, reproducible password that is uniquely different for each website (i.e., using the domain name as a seed). [EU 4.1, EU 4.2, EU 6.3] [LO 4.1.1 [P2], LO 4.2.3 [P1], LO 4.2.4 [P4], LO 6.3.1 [P1]]
  - Students will write pseudocode to describe each step of the algorithm used to generate a password. [EU 4.1] [LO 4.1.1 [P2]]
Unit 1: Computational Thinking

- Students will exchange algorithms with their peers and share feedback with each other on the clarity of the pseudocode and the strengths and weaknesses of the algorithm. [EU 4.2] [LO 4.2.4 [P4]]
- Students will construct trace tables documenting the result of each step of the algorithm in generating passwords for different domains. [EU 4.2] [LO 4.2.1 [P1], LO 4.2.4 [P4]]
- Students will write about the dangers of reusing passwords across multiple websites and online services and how such behavior may be exploited. [EU 6.3] [LO 6.3.1 [P1]]

Unit Assessments

- Minor exercises addressing specific unit topics and objectives
- Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
- Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

- Big Idea 2: Abstraction
  - Describe the different levels of abstraction for high-level and low-level programming languages and their impact on the readability of programs. [EK 2.2.3A, EK 2.2.3B] [P3]
  - Describe the hierarchical relationship between high-level and low-level programming languages in terms of programs being written by a human and executed by a computer. [EK 2.2.3C, EK 2.2.3D] [P3]
- Big Idea 4: Algorithms
  - Develop multiple algorithms for solving the same problem, identifying characteristics of the problem that lead to performance variations in different solutions. [EK 4.1.1H, EK 4.1.1I] [P2]
  - Explain how the choice of language can improve the clarity and readability of an algorithm, but not whether an algorithmic solution exists. [EK 4.1.2F, EK 4.1.2I] [P5]
  - Identify whether the number of steps required by an algorithm to solve a problem is proportional to the size of the input for the problem. [EK 4.2.1B] [P1]
  - Identify problems whose solutions can be evaluated in a reasonable time. [EK 4.2.1A] [P1]
  - Identify problems whose solutions cannot be evaluated in a reasonable time without the use of a heuristic. [EK 4.2.1C, EK 4.2.1D] [P1]
  - Explain how heuristics are used to find quick, approximate solutions to problems that are too complex to be solved in a reasonable time, such as "find the best" or "find the smallest". [EK 4.2.2A, EK 4.2.2B, EK 4.2.2C] [P1]
  - Identify problems that cannot be solved using any algorithm. [EK 4.2.2D] [P1]
Unit 1: Computational Thinking

- Identify problems that are undecidable and whose algorithms can produce a definitive answer for only some inputs. [EK 4.2.3A, EK 4.2.3C] [P1]
- Identify problems that are decidable and whose algorithms can produce a definitive answer for all inputs. [EK 4.2.3B] [P1]
- Analytically evaluate an algorithm's efficiency and correctness by reasoning formally or mathematically about the algorithm. [EK 4.2.4A, EK 4.2.4C] [P4]
- Evaluate different algorithms for the same problem in terms of their execution time, memory usage, and complexity. [EK 4.2.4D, EK 4.2.4E, EK 4.2.4G] [P4]
- Explain how an efficient algorithm for a problem can help solve larger instances of the problem. [EK 4.2.4F] [P4]
- Evaluate and compare the performance of linear search on any sorted or unsorted list with binary search on sorted lists. [EK 4.2.4H] [P4]

- Big Idea 5: Programming
  - Explain how executable programs and automation increase the scale of problems and sets of problems that can be addressed. [EK 5.2.1I, EK 5.2.1J] [P3]
  - Explain how improvements in algorithms, hardware, and software increase the kinds of problems and the size of problems solvable by programming. [EK 5.2.1K] [P3]

- Big Idea 6: The Internet
  - Identify the security tradeoffs involved in the Internet's use of the trust model in key areas, like the domain name system (DNS) or the certificate authorities (CAs) issuing of digital certificates for validating ownership of encrypted keys used in secured communication. [EK 6.3.1A, EK 6.3.1B, EK 6.3.1M] [P1]
  - Identify the software, hardware, and human components of cybersecurity. [EK 6.3.1C] [P1]
  - Explain the methods and devastating effects of various forms of cyber warfare and cybercrime, including distributed denial of service attacks (DDoS), phishing, viruses, and other attacks. [EK 6.3.1D, EK 6.3.1E, EK 6.3.1F] [P1]
  - Explain how antivirus software and firewalls can help prevent unauthorized access to private data. [EK 6.3.1G] [P1]
  - Explain how the mathematical foundation of cryptography and the use of open standards enable the functionality and security needed for effective cybersecurity. [EK 6.3.1H, EK 6.3.1I, EK 6.3.1J] [P1]
  - Explain the differences in security provided by symmetric (single-key) encryption vs. asymmetric (public key) encryption. [EK 6.3.1K, EK 6.3.1L] [P1]

- Big Idea 7: Global Impact
  - Explain how industries use Moore's law to plan future research and development. [EK 7.2.1F] [P1]
## Unit 2: Programming

### Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>2A</td>
<td>Unit Project</td>
<td>Scratch Program Project</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2B</td>
<td>Visual Programming</td>
<td>Welcome to Scratch Programming with Blocks</td>
<td>EU 1.2, EU 4.1, EU 5.1, EU 5.2, EU 7.3</td>
<td>LO 1.2.1, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1, LO 7.3.1</td>
</tr>
<tr>
<td>2C</td>
<td>Coding Skills</td>
<td>Choreographing a Dance</td>
<td>EU 4.1</td>
<td>LO 4.1.2</td>
</tr>
<tr>
<td>2D</td>
<td>Unit Project</td>
<td>Scratch Program Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2E</td>
<td>Program State</td>
<td>User Input</td>
<td>EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.2.1, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2F</td>
<td>Big Picture</td>
<td>Why Program?</td>
<td>EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 4.1.2, LO 5.1.1, LO 5.2.1</td>
</tr>
<tr>
<td>2G</td>
<td>Selection Statements</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2H</td>
<td>Quiz Show</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2I</td>
<td>Coding Skills</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2J</td>
<td>Project Presentations</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2K</td>
<td>Unit Project</td>
<td>Scratch Program Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2L</td>
<td>Coding Skills</td>
<td>Debugging</td>
<td>EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2M</td>
<td>Project Presentations</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2N</td>
<td>Coding Skills</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2O</td>
<td>Project Presentations</td>
<td>Simple &quot;if...else&quot; Statements</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2P</td>
<td>Unit Exam</td>
<td>Unit 2 Exam</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2Q</td>
<td>Unit Project</td>
<td>Scratch Program Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2R</td>
<td>Unit Exam</td>
<td>Unit 2 Exam</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
<tr>
<td>2S</td>
<td>Unit Project</td>
<td>Project Presentations</td>
<td>EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2</td>
<td>LO 1.1.1, LO 1.2.1, LO 1.2.4, LO 1.2.5, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.2.1</td>
</tr>
</tbody>
</table>
Unit 2: Programming

Unit Description

When used correctly, computational technologies can prove be extremely powerful and effective tools for solving a wide range of problems. But in order to fully harness that power, an individual needs to be proficient in instructing those tools to perform highly precise operations in well-structured and logical sequences. This unit seeks to ease students into this new, structured, and more formalized way of thinking about problem solving and programming through the use of Scratch, a block-based, visual programming language.

Once introduced to the Scratch platform and programming environment, students will then experiment with a number of basic programming concepts and constructs, such as variables, user input, and selection statements. In the process, students will not only learn how to implement intended functionality by constructing well-designed blocks of executable code, but they will also explore techniques for debugging their code and verifying its correctness.

Unit Coverage

- Big Ideas: BI 1, BI 4, BI 5, BI 7
- Enduring Understandings: EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2, EU 7.3
- Learning Objectives: LO 1.1.1 [P2], LO 1.2.1 [P2], LO 1.2.5 [P4], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3], LO 7.3.1 [P4]
- Computational Thinking Practices: P2, P3, P4, P5, P6

Unit Topics [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2a] [CR2d] [CR2e] [CR2g]

- Visual Programming [EU 1.2, EU 4.1, EU 5.1, EU 5.2, EU 7.3] [LO 1.2.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3], LO 7.3.1 [P4]]
  - Students will utilize a graphical editor to read, construct, and execute dynamic programs.
  - Students will examine, modify, and execute programs developed by others.
  - Students will share and collaborate on their own programs.
  - Students will examine how well-specified behavior of objects can be constructed through sequential actions and operations.
- Program State [EU 1.2, EU 4.1, EU 5.1, EU 5.2] [LO 1.2.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3]]
  - Students will write programs that incorporate dynamic, user-driven, keyboard controls and input.
  - Students will examine how the dynamic state of an object or program can be stored and changed using variables.
  - Students will analyze the role of clear, descriptive names for objects, behaviors, variables, and other identifiers in maintaining the readability of code.
- Selection Statements [EU 1.1, EU 1.2, EU 4.1, EU 5.1, EU 5.2] [LO 1.1.1 [P2], LO 1.2.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3]]
  - Students will examine the uses of selection statements in programming.
Unit 2: Programming

- Students will analyze the differences between simple selection and complex, nested selection statements.
- Students will examine the use of the Boolean operators "AND", "OR", and "NOT" in constructing complex conditional statements.

- Coding Skills [EU 1.2, EU 4.1, EU 5.1, EU 5.2] [LO 1.2.5 [P4], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3]]
  - Students will examine non-traditional forms of domain-specific notation.
  - Students will design and construct instructions using a non-traditional, domain-specific notation.
  - Students will analyze the clarity and legibility of instructions written in a non-traditional, domain-specific notation by reading and executing instructions created by others.
  - Students will analyze and evaluate the correctness of their programs.
  - Students will examine a number of common programming errors.
  - Students will explore a number of common debugging strategies.
  - Students will develop solutions for correcting common programming errors.

- Big Picture [EU 4.1, EU 5.1, EU 5.2] [LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.2.1 [P3]]
  - Students will examine and discuss the motivations behind a number of high-profile individuals in the field of programming.
  - Students will discuss the benefits of programming as a tool and a profession.

Unit Project [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2a] [CR2d] [CR2e]

- Scratch Program Project
  - Students will collaborate in pairs to design, implement, and debug a novel, aesthetically pleasing, and intuitive program using the Scratch programming environment. [EU 1.1, EU 1.2, EU 4.1, EU 5.1] [LO 1.1.1 [P2], LO 1.2.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.3 [P6]]
  - Students will identify a specific purpose that their program will serve (e.g., entertainment, problem solving, education, artistic expression, etc.). [EU 5.1] [LO 5.1.1 [P2]]
  - Students will integrate interactive and multimedia elements into their program. [EU 5.1] [LO 5.1.1 [P2]]
  - Students will integrate common programming constructs, such as variables and selection statements into their program. [EU 4.1, EU 5.2] [LO 4.1.2 [P5], LO 5.2.1 [P3]]
  - Students will test, debug, and correct their program. [EU 1.1, EU 1.2] [LO 1.1.1 [P2], LO 1.2.1 [P2]]
  - Students will use appropriate terminology while writing documentation detailing the full use of their program and its features. [EU 5.1] [LO 5.1.2 [P2]]
  - Students will explain their design and implementation choices while demonstrating and sharing their finished programs with their peers. [EU 1.2, EU 5.1] [LO 1.2.4 [P6], LO 5.1.3 [P6]]
Unit 2: Programming

- Students will provide a written analysis of at least one other design team's program, identifying its strengths and weaknesses and offering suggestions for improvement. [EU 1.2] [LO 1.2.4 [P6], LO 1.2.5 [P4]]

Unit Assessments

- Minor exercises addressing specific unit topics and objectives
- Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
- Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

- Big Idea 1: Creativity
  - Apply an iterative and exploratory development process to create a computational artifact using non-prescribed techniques, novel combinations of artifacts, and/or personal curiosities. [EK 1.1.1A, EK 1.1.1B] [P2]
  - Design and create a computational artifact (e.g., program, image, audio, video, presentation, etc.) for creative expression using appropriate software tools and techniques (e.g., programming IDEs, spreadsheet, 3D printer, text editor, etc.). [EK 1.2.1A, EK 1.2.1B, EK 1.2.1C, EK 1.2.1D, EK 1.2.1E] [P2]
  - Create a collaborative computational artifact that reflects the diverse talents and personal ideas of all group members. [EK 1.2.4E, EK 1.2.4F]
  - Analyze the correctness, usability, functionality, and suitability of a computational artifact in terms of the context in which it is used or perceived. [EK 1.2.5A, EK 1.2.5C, EK 1.2.5D] [P4]
  - Analyze a computational artifact for weaknesses, mistakes, and errors. [EK 1.2.5B] [P4]

- Big Idea 4: Algorithms
  - Express algorithms in a programming language for execution by a computer. [EK 4.1.2C] [P5]
  - Construct algorithms using sequencing, selection, and iteration. [EK 4.1.2G] [P5]

- Big Idea 5: Programming
  - Develop a variety of programs using methods and techniques that are appropriate for the goals of the programmer. [EK 5.1.1A] [P2]
  - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge using visual, audible, or tactile inputs and outputs. [EK 5.1.1B] [P2]
  - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge using standards or methods that differ from those used for programs developed for widespread distribution. [EK 5.1.1C] [P2]
  - Identify additional desired outcomes for a program that extend beyond its original purpose. [EK 5.1.1D] [P2]
Unit 2: Programming

○ Consult and communicate with program users to identify concerns that affect the solution to problems. [EK 5.1.2G, EK 5.1.2H] [P2]

○ Use effective communication between participants in the iterative development of a program. [EK 5.1.3C, EK 5.1.3F] [P6]

○ Use collaboration to find and correct errors with developing programs. [EK 5.1.3D] [P6]

○ Explain how algorithms are implemented using program instructions that are processed sequentially during program execution. [EK 5.2.1A, EK 5.2.1B, EK 5.2.1D] [P3]

○ Explain how program instructions may involve variables that are initialized and updated, read, and written. [EK 5.2.1C] [P3]

○ Explain how program execution automates processes. [EK 5.2.1E] [P3]

○ Explain how one or more processes may execute on one or more CPUs, using memory, input, and output. [EK 5.2.1F, EK 5.2.1G, EK 5.2.1H] [P3]

● Big Idea 7: Global Impact

○ Analyze the legal and ethical concerns of open source and licensed software, libraries, and code. [EK 7.3.1F, EK 7.3.1Q] [P4]
# Unit 3: Data Representation

## Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>3A</td>
<td>Unit Project</td>
<td>Unintend'o Controller Project</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 4.1.2, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3B</td>
<td>Binary Encoding of Information</td>
<td>Binary</td>
<td>EU 2.1, EU 3.3, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.1.2, LO 3.3.1, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3C</td>
<td>Coding Skills</td>
<td>Binary Birthday Cake</td>
<td>EU 2.1, EU 2.2, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3D</td>
<td>Unit Project</td>
<td>Unintend'o Controller Project (cont.)</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 4.1.2, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3F</td>
<td>Digital Approximations</td>
<td>Digitization</td>
<td>EU 2.2, EU 5.3</td>
<td>LO 2.2.1, LO 5.3.1</td>
</tr>
<tr>
<td>3H</td>
<td>Big Picture</td>
<td>Legality of Reselling Digital Music</td>
<td>EU 7.3</td>
<td>LO 7.3.1</td>
</tr>
<tr>
<td>3I</td>
<td>Unit Project</td>
<td>Unintend'o Controller Project (cont.)</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 4.1.2, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3J</td>
<td>Lists</td>
<td>Making a List</td>
<td>EU 3.3, EU 5.3, EU 5.5</td>
<td>LO 3.3.1, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3M</td>
<td>Processing a List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3N</td>
<td>Sorting a List</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3P</td>
<td>Unit Project</td>
<td>Unintend'o Controller Project (cont.)</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 4.1.2, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3Q</td>
<td>Unit Exam</td>
<td>Unit 3 Exam</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 3.3, EU 4.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.1.2, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 3.3.1, LO 4.1.2, LO 5.3.1, LO 5.5.1</td>
</tr>
<tr>
<td>3R</td>
<td>Unit Project</td>
<td>Project Presentations</td>
<td>EU 2.1, EU 2.2, EU 2.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5</td>
<td>LO 2.1.1, LO 2.2.1, LO 2.2.3, LO 2.3.1, LO 4.1.2, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.5.1</td>
</tr>
</tbody>
</table>
Unit Description

In order to make the most effective use of computational tools and data-driven applications, students need to have a clear awareness and sense of comfort with the diverse kinds of information that may be available for use by these programs and the various ways that information may be digitally represented, stored, and manipulated within the computer. This unit focuses on providing students with an overview of the various levels of abstraction that are used in the digital representation of discrete data and information.

Students will initially focus on the lowest levels of digital representation and storage by examining different base representations of numbers (including decimal and binary) and their application to ASCII and Unicode character encoding. Students will also explore the distinctions between analog and digital forms of representation. Finally, students will examine the characteristics of lists and the types of common use-cases for these linear, ordered collections, including traversing, searching, and sorting.

Unit Coverage

- **Big Ideas**: BI 2, BI 3, BI 4, BI 5
- **Enduring Understandings**: EU 2.1, EU 2.2, EU 2.3, EU 3.3, EU 4.1, EU 5.1, EU 5.3, EU 5.5
- **Learning Objectives**: LO 2.1.1 [P3], LO 2.1.2 [P5], LO 2.2.1 [P2], LO 2.2.3 [P3], LO 2.3.1 [P3], LO 3.3.1 [P4], LO 4.1.2 [P5], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.3.1 [P3], LO 5.5.1 [P1]
- **Computational Thinking Practices**: P1, P2, P3, P4, P5, P6

Unit Topics

- **Binary Encoding of Information [EU 2.1, EU 3.3, EU 5.3, EU 5.5] [LO 2.1.1 [P3], LO 2.1.2 [P5], LO 3.3.1 [P4], LO 5.3.1 [P3], LO 5.5.1 [P1]]**
  - Students will examine how numerical values are represented using different bases, including decimal and binary.
  - Students will explore methods of converting values from decimal to binary and binary to decimal.
  - Students will explore methods of counting in binary.
  - Students will examine the exponential relationship between the number of digits and their range of representable values.
  - Students will examine how alphanumeric characters and symbols may be represented using ASCII and Unicode character mappings.
  - Students will analyze the differences in state space between ASCII and Unicode standards.
  - Students will examine the implications of variable-width encodings (e.g., Morse code) vs. fixed-width encodings (e.g., Baudot code).
  - Students will explore how the interpretation of binary data is dependent upon its intended format and use, including base-64, bitmaps (*.BMP), plaintext (*.TXT), audio (*.MP3), etc.
Unit 3: Data Representation

- Digital Approximations [EU 2.2, EU 5.3] [LO 2.2.1 [P2], LO 5.3.1 [P3]]
  - Students will explore ways in which natural phenomena may be represented digitally.
  - Students will analyze the extent to which digital approximations accurately reflect the reality that they represent.
  - Students will analyze the differences between discrete (digital) and continuous (analog) representations of natural phenomena.
  - Students will examine the social implications of the ease with which perfect digital copies can be made.

- Lists [EU 3.3, EU 5.3, EU 5.5] [LO 3.3.1 [P4], LO 5.3.1 [P3], LO 5.5.1 [P1]]
  - Students will examine the use of lists as ordered data structures that may contain multiple values.
  - Students will investigate the use of index values to represent the position of an item in a list.
  - Students will analyze the implications of accessing an index position beyond the bounds of a list.
  - Students will investigate common operations for processing elements of a list, including searching for an element, removing an element, swapping the positions of two elements, or sorting an entire list into ascending or descending order.
  - Students will examine the implications of case-sensitivity on ordered lists of strings.

- Coding Skills [EU 2.1, EU 2.2, EU 5.3, EU 5.5] [LO 2.1.1 [P3], LO 2.2.1 [P2], LO 5.3.1 [P3], LO 5.5.1 [P1]]
  - Students will construct a Scratch program that simulates candles on a birthday cake being lit so as to show the user's age in binary.

- Big Picture [EU 7.3] [LO 7.3.1 [P4]]
  - Students will examine and discuss the legality of reselling "used" digital music.

Unit Project [CR1a] [CR1b] [CR1c] [CR1e] [CR1f] [CR2b] [CR2d] [CR2e]

- Unintent'o Controller Project
  - Students will develop a Scratch program that acts as a device driver for a video game controller interface. [EU 2.2, EU 2.3] [LO 2.2.1 [P2], LO 2.2.3 [P3], LO 2.3.1 [P3]]
  - Students will map each of six controls (UP, DOWN, LEFT, RIGHT, A, and B) to individual bits. [EU 2.1, EU 2.2, EU 2.3, EU 5.3, EU 5.5] [LO 2.1.1 [P3], LO 2.2.1 [P2], LO 2.3.1 [P3], LO 5.3.1 [P3], LO 5.5.1 [P1]]
  - Students will map each binary pattern of button presses to different game actions (e.g., walk forward, walk backward, turn left, turn right, jump, duck, whirl, leap, crawl, etc.). [EU 2.1, EU 2.2, EU 2.3, EU 5.3] [LO 2.1.1 [P3], LO 2.2.1 [P2], LO 2.3.1 [P3], LO 5.3.1 [P3]]
Unit 3: Data Representation

- Students will write detailed specifications and justifications for each button-to-action mapping of their design. [EU 2.1, EU 2.2, EU 4.1, EU 5.1, EU 5.3] [LO 2.1.1 [P3], LO 2.2.3 [P3], LO 4.1.2 [P5], LO 5.1.2 [P2], LO 5.3.1 [P3]]
- Students will collaborate with their peers throughout the design and development process to determine end-user requests for features and to share feedback on design and implementation strategies. [EU 5.1] [LO 5.1.3 [P6]]
- Students will write documentation detailing the use of their program and its features using appropriate terminology. [EU 5.1] [LO 5.1.2 [P2]]

Unit Readings


Unit Assessments

- Minor exercises addressing specific unit topics and objectives
- Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
- Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

- Big Idea 2: Abstraction
  - Describe how digital data is represented by abstractions at different levels, including bits at the lowest level and numbers, characters, color, etc. at the higher levels. [EK 2.1.1A, EK 2.1.1B, EK 2.1.1C] [P3]
  - Describe the methods and reasons for representing numbers in different bases (e.g., binary/base-2, decimal/base-10, hexadecimal/base-16, etc.) and how to convert between them. [EK 2.1.1D, EK 2.1.1E, EK 2.1.1F, EK 2.1.1G] [P3]
  - Explain how finite binary sequences can be used to represent different types of data and instructions, depending on context. [EK 2.1.2A, EK 2.1.2D, EK 2.1.2E, EK 2.1.2F] [P5]
  - Explain how integer representations can result in "overflow" and "underflow" errors for values that exceed the range allowed by a fixed number of bits. [EK 2.1.2B] [P5]
  - Explain how real number representations can result in "round-off" errors for values that exceed the precision allowed by a fixed number of bits. [EK 2.1.2C] [P5]
  - Develop an abstraction by identifying common features and removing detail in order to generalize concepts and functionality. [EK 2.2.1A, EK 2.2.1B] [P2]
  - Develop an abstraction that uses parameters to enable the reuse of generalized software functionality. [EK 2.2.1C] [P2]
Unit 3: Data Representation

○ Explain how binary data is processed by physical layers of computing hardware, including gates, chips, and components. [EK 2.2.3E] [P3]

○ Describe the hierarchical relationship between the different levels of abstraction in computer hardware, including high-level components (e.g. video cards) and low-level components (e.g., chips, circuits, transistors, and gates). [EK 2.2.3F, EK 2.2.3G, EK 2.2.3H, EK 2.2.3I] [P3]

○ Explain how applications and systems are designed, developed, and analyzed using lower-level hardware, software, and conceptual abstractions and combining them to form higher-level abstractions. [EK 2.2.3J, EK 2.2.3K] [P3]

○ Use various levels of abstraction to construct a model or simulation that omits unnecessary features in order to create a simplified representation that mimics real-world events without the cost or danger of building and testing the phenomena in the real world. [EK 2.3.1A, EK 2.3.1B, EK 2.3.1C, EK 2.3.1D] [P3]

- Big Idea 3: Data and Information
  ○ Analyze how the characteristics of data, the methods and costs of manipulating the data, and the intended uses of data relate to the storage requirements and choice of storage media. [EK 3.3.1G, EK 3.3.1H, EK 3.3.1I] [P4]

- Big Idea 4: Algorithms
  ○ Express algorithms in natural language and pseudocode for human readability. [EK 4.1.2B] [P5]

- Big Idea 5: Programming
  ○ Provide documentation about program components, such as blocks and procedures, to maintain correct programs when working individually or collaboratively with other programmers. [EK 5.1.2D, EK 5.1.2E, EK 5.1.2F] [P2]
  ○ Consult and communicate with program users to identify concerns that affect the solution to problems. [EK 5.1.2G, EK 5.1.2H] [P2]
  ○ Use collaboration to facilitate multiple perspectives in developing ideas for solving problems by programming. [EK 5.1.3B] [P6]
  ○ Use effective communication between participants in the iterative development of a program. [EK 5.1.3C, EK 5.1.3F] [P6]
  ○ Employ data abstraction and its ability to separate behavior from implementation by using a variety of abstract data types, including strings, integers, floating-point numbers, and lists. [EK 5.3.1H, EK 5.3.1I, EK 5.3.1J, EK 5.3.1K] [P3]
  ○ Use lists and procedures as abstractions in programming to produce programs that are easier to develop and maintain. [EK 5.3.1L] [P3]
  ○ Explain how numbers and numerical concepts are expressed in programming as integers (finitely bound by storage limitations) and real numbers (approximated with limited precision). [EK 5.5.1A, EK 5.5.1B, EK 5.5.1C] [P1]
  ○ Construct mathematical and logical expressions using arithmetic and Boolean operators. [EK 5.5.1D, EK 5.5.1E, EK 5.5.1F] [P1]
Unit 3: Data Representation

- Employ intuitive and formal reasoning about program components using Boolean concepts. [EK 5.5.1G] [P1]
- Employ lists and collections as abstract data types (ADTs) that provide functionality to add, remove, and iterate over all elements, as well as to determine whether an element is in a collection. [EK 5.5.1H, EK 5.5.1I, EK 5.5.1J] [P1]
## Unit 4: Digital Media Processing

### Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>4A</td>
<td>Unit Project</td>
<td>Image Filter Project</td>
<td>EU 1.2, EU 1.3, EU 3.3, EU 5.1, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1</td>
</tr>
<tr>
<td>4B</td>
<td>Procedural Programming</td>
<td>Introduction to Processing</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4C</td>
<td>Procedural Programming</td>
<td>Drawing</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4D</td>
<td>Procedural Programming</td>
<td>Mouse Interaction</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4E</td>
<td>Procedural Programming</td>
<td>Keyboard Interaction</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4F</td>
<td>Big Picture</td>
<td>Ethics of Digital Manipulation</td>
<td>EU 7.3</td>
<td>LO 7.3.1</td>
</tr>
<tr>
<td>4G</td>
<td>Image Manipulation</td>
<td>RGB Color</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4H</td>
<td>Image Manipulation</td>
<td>Raster Images</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4I</td>
<td>Image Manipulation</td>
<td>Raster Image Manipulation</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4J</td>
<td>Image Manipulation</td>
<td>Encoding Schemes</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4K</td>
<td>Image Manipulation</td>
<td>Manipulating Digital Images</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.3.1, LO 2.2.2, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 5.3.1, LO 5.4.1</td>
</tr>
<tr>
<td>4L</td>
<td>Unit Project</td>
<td>Image Filter Project (cont.)</td>
<td>EU 1.2, EU 1.3, EU 3.3, EU 5.1, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1</td>
</tr>
<tr>
<td>4M</td>
<td>Audio Manipulation</td>
<td>Digital Audio</td>
<td>EU 1.3, EU 3.3, EU 4.1, EU 5.1, EU 7.3</td>
<td>LO 1.3.1, LO 3.3.1, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 7.3.1</td>
</tr>
<tr>
<td>4N</td>
<td>Audio Manipulation</td>
<td>Audio Processing</td>
<td>EU 1.3, EU 3.3, EU 4.1, EU 5.1, EU 7.3</td>
<td>LO 1.3.1, LO 3.3.1, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 7.3.1</td>
</tr>
<tr>
<td>4O</td>
<td>Audio Manipulation</td>
<td>Audio Compression</td>
<td>EU 1.3, EU 3.3, EU 4.1, EU 5.1, EU 7.3</td>
<td>LO 1.3.1, LO 3.3.1, LO 4.1.1, LO 4.1.2, LO 5.1.2, LO 7.3.1</td>
</tr>
<tr>
<td>4P</td>
<td>Big Picture</td>
<td>Intellectual Property Rights</td>
<td>EU 7.3</td>
<td>LO 7.3.1</td>
</tr>
<tr>
<td>4Q</td>
<td>Unit Project</td>
<td>Image Filter Project (cont.)</td>
<td>EU 1.2, EU 1.3, EU 3.3, EU 5.1, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1</td>
</tr>
<tr>
<td>4R</td>
<td>Unit Project</td>
<td>Image Filter Project (cont.)</td>
<td>EU 1.2, EU 1.3, EU 3.3, EU 5.1, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1</td>
</tr>
<tr>
<td>4S</td>
<td>Unit Exam</td>
<td>Unit 4 Exam</td>
<td>EU 1.2, EU 1.3, EU 2.2, EU 3.3, EU 4.1, EU 5.1, EU 5.3, EU 5.4, EU 7.3</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1, LO 7.3.1</td>
</tr>
<tr>
<td>4T</td>
<td>Unit Project</td>
<td>Project Presentations</td>
<td>EU 1.2, EU 1.3, EU 3.3, EU 5.1, EU 5.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 1.3.1, LO 3.3.1, LO 5.1.2, LO 5.1.3, LO 5.4.1</td>
</tr>
</tbody>
</table>
Unit 4: Digital Media Processing

Unit Description

Building upon their earlier, visual programming experiences with Scratch, this unit guides students through the transition to programming in a high-level, procedural language through a brief an introduction to Processing. By familiarizing themselves with a text-based environment that more closely reflects the actual programming tools used in industry, such as Java, C++, or Python, students will be better equipped for continuing their studies in computer science beyond the scope of this course.

With the help of Processing's graphical programming model that is designed to simplify the task of creating sophisticated, visual artifacts, students will explore the characteristics of the RGB color model and its use in encoding digital images. For the unit project, they will apply these concepts toward the implementation of a series of algorithmic filters for digitally modifying images to achieve various visual effects. Finally, students will also investigate the methods of representing and modifying digital audio, including Auto-Tune and audio compression.

Unit Coverage

- Big Ideas: BI 1, BI 2, BI 3, BI 4, BI 5, BI 7
- Enduring Understandings: EU 1.2, EU 1.3, EU 2.2, EU 3.3, EU 4.1, EU 5.1, EU 5.3, EU 5.4, EU 7.3
- Learning Objectives: LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 1.3.1 [P2], LO 2.2.2 [P3], LO 3.3.1 [P4], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.3.1 [P3], LO 5.4.1 [P4], LO 7.3.1 [P4]
- Computational Thinking Practices: P2, P3, P4, P5, P6

Unit Topics

- Procedural Programming [EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4] [LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.3.1 [P2], LO 2.2.2 [P3], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 5.1.2 [P2], LO 5.3.1 [P3], LO 5.4.1 [P4]]
  - Students will explore the capabilities of a text-based programming language (Processing).
  - Students will compare and contrast the programming capabilities of a visual programming language (Scratch) with those of a text-based programming language (Processing).
  - Students will write programs that make use of parameterized methods to invoke specific behaviors.
  - Students will understand the importance of using proper punctuation and syntax when coding in a text-based programming language.
  - Students will use event handlers to animate on-screen effects and respond to mouse and keyboard input.
  - Students will write code using common programming constructs like conditional if() for selection and while() loops for iteration.
Unit 4: Digital Media Processing

APPLICATION

- Image Manipulation [EU 1.2, EU 1.3, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4] [LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 1.3.1 [P2], LO 2.2.2 [P3], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 5.1.2 [P2], LO 5.3.1 [P3], LO 5.4.1 [P4]]
  - Students will examine the structure of raster images as compositions of individual pixels.
  - Students will explore various methods of representing color, including RGB, CMYK, and HSV.
  - Students will explore the various colors that can be produced by the combination of different ratios of red, green, and blue light.
  - Students will modify the color channels of pixels in an image to produce a variety of effects.
  - Students will design algorithms for modifying the pixels in an image in prescribed ways to create custom image filters.
  - Students will explore the difference between lossy and lossless encoding schemes of several common image file formats.

- Audio Manipulation [EU 1.3, EU 3.3, EU 4.1, EU 5.1, EU 7.3] [LO 1.3.1 [P2], LO 3.3.1 [P4], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 5.1.2 [P2], LO 7.3.1 [P4]]
  - Students will analyze the differences between analog and digital sound.
  - Students will explore the roles that sampling rate and bit depth play in determining the quality of digitized sound.
  - Students will explore methods of programmatically generating digital audio.
  - Students will explore methods of programmatically altering and modifying digital audio by adjusting volume, pitch, and sampling rate.
  - Students will explore the methods and effects of compression algorithms in reducing the amount of data needed to represent an audio sample.

- Big Picture [EU 7.3] [LO 7.3.1 [P4]]
  - Students will explore the positive and negative consequences of digitally altering images.
  - Students will discuss the ethics of digitally manipulating images, especially in the context of journalism.
  - Students will discuss the issues related to intellectual property.
  - Students will explore the limitations and rights associated with a number of common licenses, including Creative Commons.

Unit Project [CR1b] [CR1f] [CR2a] [CR2e]

- Image Filter Project
  - Students will utilize pair programming to design and implement a program for filtering digital images. [EU 1.2, EU 3.3, EU 5.1] [LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 3.3.1 [P4], LO 5.1.2 [P2], LO 5.1.3 [P6]]
  - Using the Processing programming language, students will develop code to systematically transform an image by mathematically manipulating its bits, pixel
Unit 4: Digital Media Processing

APPLICATION

by pixel. [EU 1.2, EU 1.3, EU 3.3, EU 5.1] [LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.3.1 [P2], LO 3.3.1 [P4], LO 5.1.2 [P2]]
○ Students will write documentation detailing the use of their program and its features using appropriate terminology. [EU 5.1, EU 5.4] [LO 5.1.2 [P2], LO 5.4.1 [P4]]
○ Students will explain their design and implementation choices by demonstrating and sharing their finished programs with their peers. [EU 1.2, EU 5.1] [LO 1.2.4 [P6], LO 5.1.3 [P6]]

Unit Readings

● Blown to Bits (Abelson, Ledeen, Lewis). Chapter 6: Balance Toppled – Who Owns the Bits?

Unit Assessments

● Minor exercises addressing specific unit topics and objectives
● Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
● Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

● Big Idea 1: Creativity
  ○ Create a computational artifact using computing tools and innovative, non-traditional techniques to solve a problem. [EK 1.2.2A, EK 1.2.2B] [P2]
  ○ Create a computational artifact by combining and modifying existing artifacts to show personal expression of ideas. [EK 1.2.3A, EK 1.2.3C] [P2]
  ○ Use computational tools to create or modify a computational artifact with enhanced detail and precision. [EK 1.2.3B] [P2]
  ○ Use appropriate interpersonal skills, communication, and group decision-making to create an enhanced, collaborative computational artifact. [EK 1.2.4C, EK 1.2.4D] [P6]
  ○ Create a collaborative computational artifact that reflects the diverse talents and personal ideas of all group members. [EK 1.2.4E, EK 1.2.4F] [P6]
  ○ Identify how the creation of digital effects, images, audio, video, and animations has transformed industries. [EK 1.3.1A] [P2]
  ○ Use computing tools to create digital audio and music by synthesizing, sampling, recording, layering, and/or looping sounds. [EK 1.3.1B] [P2]
  ○ Use computing tools to create digital images by generating pixel patterns, manipulating digital images, or combining images. [EK 1.3.1C] [P2]
  ○ Use appropriate software and image editing tools to create digital effects and animations. [EK 1.3.1D] [P2]
Unit 4: Digital Media Processing

○ Use computing tools to enable creative exploration of digital images and/or sounds. [EK 1.3.1E] [P2]

● Big Idea 2: Abstraction
  ○ Develop software using multiple levels of abstraction, including constants, expressions, statements, procedures, and libraries, to more effectively apply available resources and tools to solve problems. [EK 2.2.2A, EK 2.2.2B] [P3]

● Big Idea 3: Data and Information
  ○ Analyze the different trade-offs between lossy and lossless compression techniques for storing and transmitting data. [EK 3.3.1C, EK 3.3.1D, EK 3.3.1E] [P4]

● Big Idea 4: Algorithms
  ○ Develop an algorithm using sequencing, selection, and iteration. [EK 4.1.1A, EK 4.1.1B, EK 4.1.1C, EK 4.1.1D] [P2]
  ○ Develop an algorithm that uses or combines existing, standard algorithms to ensure correctness of the resulting solution. [EK 4.1.1E, EK 4.1.1F, EK 4.1.1G] [P2]
  ○ Explain how natural language, pseudocode, and visual and textual programming languages can all be used to express an algorithm. [EK 4.1.2A, EK 4.1.2H] [P5]
  ○ Explain how different languages are better suited than others for expressing algorithms in specific problem domains. [EK 4.1.2D, EK 4.1.2E] [P5]

● Big Idea 5: Programming
  ○ Identify ways that advances in computing have enabled creativity in other fields. [EK 5.1.1F] [P2]
  ○ Develop a large, correct program using an iterative process that incrementally combines tested program components. [EK 5.1.2A, EK 5.1.2B, EK 5.1.2C] [P2]
  ○ Provide documentation about program components, such as blocks and procedures, to maintain correct programs when working individually or collaboratively with other programmers. [EK 5.1.2D, EK 5.1.2E, EK 5.1.2F] [P2]
  ○ Develop a program using appropriate knowledge of and skill in the development process, including designing, implementing, testing, debugging, and maintaining programs. [EK 5.1.2I, EK 5.1.2J] [P2]
  ○ Use collaboration to facilitate multiple perspectives in developing ideas for solving problems by programming. [EK 5.1.3B] [P6]
  ○ Use abstraction to create named, parameterized, and reusable blocks of programming in order to reduce the complexity of writing and maintaining a program. [EK 5.3.1A, EK 5.3.1B, EK 5.3.1C, EK 5.3.1D] [P3]
  ○ Use parameterization to generalize specific solutions and allow a single function to be used in place of duplicated code. [EK 5.3.1E, EK 5.3.1F, EK 5.3.1G] [P3]
  ○ Use well-documented application program interfaces (APIs) and libraries to connect software components and to simplify complex programming. [EK 5.3.1M, EK 5.3.1N, EK 5.3.1O] [P3]
  ○ Use good programming style, such as meaningful names for variables and procedures, shorter code blocks, and non-duplicated code, in order to improve
the determination of program correctness. [EK 5.4.1A, EK 5.4.1B, EK 5.4.1C, EK 5.4.1D] [P4]
○ Debug a program by locating and correcting errors. [EK 5.4.1E] [P4]
○ Describe the functionality of a program at a high level in terms of what it does and how a user interacts with it and provide examples of intended behavior on specific inputs in order to find program errors. [EK 5.4.1F, EK 5.4.1G, EK 5.4.1L, EK 5.4.1M, EK 5.4.1N] [P4]
○ Use visual displays (or different modalities) of program state to help in finding errors. [EK 5.4.1H] [P4]
○ Justify a program's correctness by explaining how it meets its specifications. [EK 5.4.1I, EK 5.4.1J] [P4]
○ Demonstrate the correctness of a program by demonstrating correctness of its components, including code blocks and procedures. [EK 5.4.1K] [P4]

● Big Idea 7: Global Impact
○ Analyze the legal and ethical concerns raised by innovations, access, and censorship of digital content. [EK 7.3.1A, EK 7.3.1B, EK 7.3.1C, EK 7.3.1D, EK 7.3.1E] [P4]
○ Analyze the intellectual property and copyright concerns with digital information, audio, video, and textual content. [EK 7.3.1N, EK 7.3.1O, EK 7.3.1P] [P4]
## Unit 5: Big Data

### Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>5A</td>
<td>Unit Project</td>
<td>TEDxKinda Project</td>
<td>EU 1.2, EU 2.3, EU 3.1, EU 3.2, EU 5.1, EU 7.1, EU 7.2, EU 7.3, EU 7.5</td>
<td>LO 1.2.4, LO 2.3.2, LO 3.1.1, LO 3.1.2, LO 3.1.3, LO 3.2.1, LO 3.2.2, LO 5.1.1, LO 5.1.3, LO 7.1.2, LO 7.2.1, LO 7.3.1, LO 7.5.1, LO 7.5.2</td>
</tr>
<tr>
<td>5B</td>
<td>Data Science</td>
<td>Introduction to Big Data</td>
<td>EU 3.1, EU 3.2, EU 5.1, EU 7.1, EU 7.2</td>
<td>LO 3.1.1, LO 3.1.3, LO 3.2.1, LO 5.1.1, LO 7.1.2, LO 7.2.1</td>
</tr>
<tr>
<td>5C</td>
<td>Coding Skills</td>
<td>Automatic Summarization</td>
<td>EU 3.1, EU 3.2, EU 5.1</td>
<td>LO 3.1.1, LO 3.1.3, LO 3.2.1, LO 5.1.1, LO 7.1.2, LO 7.2.1</td>
</tr>
<tr>
<td>5D</td>
<td>Data Aggregation</td>
<td>Collection</td>
<td>EU 3.1, EU 3.2, EU 3.3</td>
<td>LO 3.1.1, LO 3.2.1, LO 3.2.2, LO 3.3.1</td>
</tr>
<tr>
<td>5F</td>
<td>Unit Project</td>
<td>TEDxKinda Project (cont.)</td>
<td>EU 1.2, EU 2.3, EU 3.1, EU 3.2, EU 5.1, EU 7.1, EU 7.2, EU 7.3, EU 7.5</td>
<td>LO 1.2.4, LO 2.3.2, LO 3.1.1, LO 3.1.2, LO 3.1.3, LO 3.2.1, LO 3.2.2, LO 5.1.1, LO 5.1.3, LO 7.1.2, LO 7.2.1, LO 7.3.1, LO 7.5.1, LO 7.5.2</td>
</tr>
<tr>
<td>5G</td>
<td>Data Aggregation</td>
<td>Extraction</td>
<td>EU 3.1, EU 3.2, EU 3.3</td>
<td>LO 3.1.1, LO 3.1.3, LO 3.2.1, LO 3.2.2, LO 3.3.1</td>
</tr>
<tr>
<td>5H</td>
<td>Big Picture</td>
<td>Data Breaches</td>
<td>EU 3.2, EU 3.3, EU 5.1, EU 7.3</td>
<td>LO 3.2.2, LO 3.3.1, LO 5.1.1, LO 7.3.1</td>
</tr>
<tr>
<td>5J</td>
<td>Data Analysis</td>
<td>Statistical Analysis</td>
<td>EU 2.3, EU 3.1, EU 3.2, EU 4.2, EU 7.2</td>
<td>LO 2.3.2, LO 3.1.1, LO 3.1.3, LO 3.2.1, LO 3.2.2, LO 4.2.4, LO 7.2.1</td>
</tr>
<tr>
<td>5K</td>
<td>Clustering and Classification</td>
<td>Data Mining</td>
<td>EU 3.1, EU 3.2, EU 7.1, EU 7.2</td>
<td>LO 3.1.1, LO 3.1.2, LO 3.2.1, LO 3.2.2, LO 7.1.2, LO 7.2.1</td>
</tr>
<tr>
<td>5L</td>
<td>Big Picture</td>
<td>Crowdsourcing</td>
<td>EU 3.1, EU 3.2, EU 7.1, EU 7.2</td>
<td>LO 3.1.1, LO 3.1.2, LO 3.2.1, LO 3.2.2, LO 7.1.2, LO 7.2.1</td>
</tr>
<tr>
<td>5M</td>
<td>Unit Project</td>
<td>TEDxKinda Project (cont.)</td>
<td>EU 1.2, EU 2.3, EU 3.1, EU 3.2, EU 5.1, EU 7.1, EU 7.2, EU 7.3, EU 7.5</td>
<td>LO 1.2.4, LO 2.3.2, LO 3.1.1, LO 3.1.2, LO 3.1.3, LO 3.2.1, LO 3.2.2, LO 5.1.1, LO 5.1.3, LO 7.1.2, LO 7.2.1, LO 7.3.1, LO 7.5.1, LO 7.5.2</td>
</tr>
</tbody>
</table>
Unit 5: Big Data

<table>
<thead>
<tr>
<th></th>
<th>APPLICATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>5S</td>
<td>Unit Exam</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>5T</td>
<td>Unit Project</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Unit Description

One of the most powerful applications of computational thinking relates to the creation and analysis of large data sets. In this unit, students will explore the complete set of processes and techniques that are involved in collecting large volumes of raw data and extracting new and useful information. Students will look at a variety of ways that data scientists use techniques such as statistical analysis, data mining, clustering, classification, and automatic summarization to construct and visualize new knowledge. And finally, using these techniques themselves, students will perform their own analysis on a sample data set to discover new insights, which they will share with the class through a formal, TED-style presentation.

Unit Coverage

- Big Ideas: BI 1, BI 2, BI 3, BI 4, BI 5, BI 7
- Enduring Understandings: EU 1.2, EU 2.3, EU 3.1, EU 3.2, EU 3.3, EU 4.2, EU 5.1, EU 7.1, EU 7.2, EU 7.3
- Learning Objectives: LO 1.2.4 [P6], LO 2.3.2 [P3], LO 3.1.1 [P4], LO 3.1.2 [P6], LO 3.1.3 [P5], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 3.3.1 [P4], LO 4.2.4 [P4], LO 5.1.1 [P2], LO 5.1.3 [P6], LO 7.1.2 [P4], LO 7.2.1 [P1], LO 7.3.1 [P4]
- Computational Thinking Practices: P1, P2, P3, P4, P5, P6

Unit Topics [CR1a] [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2b] [CR2c] [CR2d] [CR2e] [CR2g]

- Data Science [EU 3.1, EU 3.2, EU 5.1, EU 7.1, EU 7.2] [LO 3.1.1 [P4], LO 3.1.3 [P5], LO 3.2.1 [P1], LO 5.1.1 [P2], LO 7.1.2 [P4], LO 7.2.1 [P1]]
  - Students will relate the impact of computing to ubiquitous and large-scale data processing.
  - Students will explore the ways that patterns within large data sets can be used in a predictive manner.
  - Students will discuss the risks and benefits of drawing conclusions from patterns found in large data sets.
Unit 5: Big Data

○ Students will combine visuals, content knowledge, and interaction to create a dynamic infographic that clearly communicates discrete information about a data set.
○ Students will identify the characteristics that differentiate usable data from unusable data.
○ Students will identify the characteristics that differentiate useful data from useless data.

● Data Aggregation [EU 3.1, EU 3.2, EU 3.3] [LO 3.1.1 [P4], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 3.3.1 [P4]]
  ○ Students will explore the purposes of various processing tasks, including collection, knowledge extraction, and data storage.
  ○ Students will identify multiple techniques for data collection, both on and off of the Internet.
  ○ Students will analyze the characteristics of structured and unstructured data.
  ○ Students will extract structured information from unstructured data.
  ○ Students will examine methods of extracting information from online sources, including structured and unstructured search engines, screen scrapers, and spiders.
  ○ Students will explore the basic features and functionality of modern relational databases.
  ○ Students will debate the implications of large-scale data storage and data persistence on privacy and utility, including the costs associated with each.

● Data Analysis [EU 2.3, EU 3.1, EU 3.2, EU 4.2, EU 7.2] [LO 2.3.2 [P3], LO 3.1.1 [P4], LO 3.1.3 [P5], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 4.2.4 [P4], LO 7.2.1 [P1]]
  ○ Students will analyze the tradeoff of utility and confidence in descriptive, predictive, and prescriptive data analysis.
  ○ Students will investigate traditional statistical hypothesis testing and exploratory data analysis.
  ○ Students will investigate the use of data mining in the discovery of patterns in large data sets.
  ○ Students will examine the use of cluster analysis and data classification in the processing of large data sets.

● Coding Skills [EU 3.1, EU 3.2, EU 5.1] [LO 3.1.1 [P4], LO 3.1.3 [P5], LO 3.2.1 [P1], LO 5.1.1 [P2]]
  ○ Students will use automatic summarization tools to create computer-generated summaries of a large data set.

● Big Picture [EU 3.1, EU 3.2, EU 3.3, EU 5.1, EU 7.1, EU 7.2, EU 7.3] [LO 3.1.1 [P4], LO 3.1.2 [P6], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 3.3.1 [P4], LO 5.1.1 [P2], LO 7.1.2 [P4], LO 7.2.1 [P1], LO 7.3.1 [P4]]
  ○ Students will examine the security risks and responsibilities assumed by companies that collect and store sensitive personal data.
  ○ Students will examine the causes and impact of data breaches involving sensitive personal data.
Unit 5: Big Data

APPLICATION

○ Students will apply the technique of crowdsourcing to a novel data collection problem.

Unit Project [CR1a] [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2a] [CR2b] [CR2c] [CR2e] [CR2g]

● TEDxKinda Project
  ○ Students will collaborate in groups to analyze public data sets and extract insightful information and new knowledge using a number of big data analysis techniques and tools. [EU 1.2, EU 2.3, EU 3.1, EU 3.2, EU 5.1, EU 7.1] [LO 1.2.4 [P6], LO 2.3.2 [P3], LO 3.1.1 [P4], LO 3.1.3 [P5], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 5.1.3 [P6], LO 7.1.2 [P4]]
  ○ Students will evaluate and justify the appropriateness of their chosen data set(s). [EU 3.1, EU 3.2, EU 7.1, EU 7.2, EU 7.3, EU 7.5] [LO 3.1.2 [P6], LO 3.2.1 [P1], LO 3.2.2 [P3], LO 7.1.2 [P4], LO 7.2.1 [P1], LO 7.3.1 [P4], LO 7.5.2 [P5]]
  ○ Students will construct informative and aesthetically pleasing data visualizations. [EU 1.2, EU 3.1] [LO 1.2.4 [P6], LO 3.1.1 [P4], LO 3.1.3 [P5]]
  ○ Students will write a script and prepare speaker notes for a formal presentation of their findings. [EU 3.1] [LO 3.1.3 [P5]]
  ○ Students will cite all online and print sources used in their research and presentation preparation. [EU 7.5] [LO 7.5.1 [P1]]
  ○ Students will deliver a TED-style presentation discussing their data analysis and findings using appropriate terminology. [EU 5.1, EU 7.1] [LO 5.1.1 [P2], LO 7.1.2 [P4]]

Unit Readings

● Blown to Bits (Abelson, Ledeen, Lewis). Chapter 2: Naked in the Sunlight – Privacy Lost, Privacy Abandoned

Unit Assessments

● Minor exercises addressing specific unit topics and objectives
● Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
● Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

● Big Idea 1: Creativity
  ○ Use appropriate collaboration tools and techniques to create a computational artifact. [EK 1.2.4A, EK 1.2.4B] [P6]
● Big Idea 2: Abstraction
Unit 5: Big Data

- Use models and simulations to form and refine hypotheses and generate new knowledge about the objects or phenomena being modeled. [EK 2.3.2A, EK 2.3.2B, EK 2.3.2C, EK 2.3.2D] [P3]
- Use simulations to test hypotheses without the constraints of the real world. [EK 2.3.2E] [P3]
- Use extensive and rapid testing of models to accurately reflect the objects or phenomena being modeled. [EK 2.3.2F, EK 2.3.2H] [P3]
- Design simulations that are appropriate for the time and resource constraints of the phenomena being modeled. [EK 2.3.2G] [P3]

● Big Idea 3: Data and Information
- Use computers in an iterative and interactive way to process digital information and gain insight and knowledge. [EK 3.1.1A] [P4]
- Use computational processes to filter and clean up digital information. [EK 3.1.1B] [P4]
- Use computers to process information through the combining of data sources and the clustering and classification of data. [EK 3.1.1C] [P4]
- Use computational tools to translate and transform digitally represented information to reveal patterns within the data. [EK 3.1.1D, EK 3.1.1E] [P4]
- Use collaboration to share multiple perspectives, experiences, and skill sets to generate greater insight and knowledge than can be obtained when working alone. [EK 3.1.2A, EK 3.1.2B, EK 3.1.2F] [P6]
- Use face-to-face and online collaborative tools on data-driven problems to facilitate processing information and generating greater insight and knowledge. [EK 3.1.2C, EK 3.1.2E] [P6]
- Use collaboration to develop and test hypotheses and answer questions in order to gain greater insight and knowledge. [EK 3.1.2D] [P6]
- Use appropriate visualization tools and software to communicate information about data via tables, diagrams, and textual displays. [EK 3.1.3A, EK 3.1.3B] [P5]
- Use summarization, transformation of information, and interactivity to communicate insight and knowledge gained from data. [EK 3.1.3C, EK 3.1.3D, EK 3.1.3E] [P5]
- Identify the challenges for extracting information and the opportunities for identifying trends from large data sets. [EK 3.2.1A, EK 3.2.1B] [P1]
- Use appropriate search and filtering tools to efficiently find and make connections with information in large data sets. [EK 3.2.1C, EK 3.2.1D, EK 3.2.1E] [P1]
- Use appropriate software tools, such as spreadsheets and databases, to efficiently organize and find trends in information. [EK 3.2.1F] [P1]
- Use metadata to add descriptive information about the organization or contents of an image, a Web page, or other complex objects in order to increase the searchability or usefulness of the data. [EK 3.2.1G, EK 3.2.1H, EK 3.2.1I] [P1]
- Use large data sets to store, retrieve, and computationally process information such as transactions, measurements, text, sound, images, and video. [EK 3.2.2A] [P3]
Unit 5: Big Data

APPLICATION

○ Identify the challenges of structuring, storing, processing and curating large data sets. [EK 3.2.2B, EK 3.2.2C] [P3]
○ Identify the challenges of maintaining privacy with large data sets that contain personal information. [EK 3.2.2D] [P3]
○ Analyze the way that the size of a data set affects its scalability and the computational and analytical techniques required to effectively store, manage, transmit, and process data. [EK 3.2.2E, EK 3.2.2F, EK 3.2.2G, EK 3.2.2H] [P3]
○ Analyze how security and privacy concerns involve trade-offs and impact the methods of storing and transmitting information. [EK 3.3.1A, EK 3.3.1B, EK 3.3.1E] [P4]

● Big Idea 4: Algorithms
○ Empirically evaluate an algorithm by implementing the algorithm and running it on different inputs. [EK 4.2.4B] [P4]

● Big Idea 5: Programming
○ Explain how 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. [EK 5.1.1E] [P2]
○ Use collaboration to decrease the size and complexity of tasks required of individual programmers and to develop program components independently. [EK 5.1.3A, EK 5.1.3E] [P6]

● Big Idea 7: Global Impact
○ Explain how the advantages of distributed computing and crowdsourcing affect the ability to solve large-scale problems related to digital data (e.g., "citizen science", SETI@Home, Amazon's Mechanical Turk, etc.). [EK 7.1.2A, EK 7.1.2B, EK 7.1.2C, EK 7.1.2D, EK 7.1.2E, EK 7.1.2F] [P4]
○ Explain how new technologies and applications are made possible by the proliferation of always-on mobile computers. [EK 7.1.2G] [P4]
○ Explain how machine learning and data mining have enabled innovations in medicine, business, and science. [EK 7.2.1A] [P1]
○ Explain how computing enables innovation and creativity in scientific and other fields. [EK 7.2.1B, EK 7.2.1C, EK 7.2.1G] [P1]
○ Explain how open access to digital information and scientific databases have benefited scientific researchers. [EK 7.2.1D, EK 7.2.1E] [P1]
○ Analyze the privacy and security concerns related to the collection, aggregation, and use of personal data. [EK 7.3.1G, EK 7.3.1H, EK 7.3.1I, EK 7.3.1J, EK 7.3.1K, EK 7.3.1L, EK 7.3.1M] [P4]
○ Use online databases and libraries to access information. [EK 7.5.1A] [P1]
○ Use advanced search tools, Boolean logic, and key words to focus or limit searches to desired results. [EK 7.5.1B] [P1]
○ Avoid plagiarism by appropriately acknowledging sources. [EK 7.5.1C] [P1]
○ Evaluate the credibility and relevance of sources of information. [EK 7.5.2A, EK 7.5.2B] [P5]
## Unit 6: Innovative Technologies

### Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>6A</td>
<td>Unit Project</td>
<td>Future Technology Project</td>
<td>EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 1.1.1, LO 1.2.4, LO 5.1.1, LO 5.1.3, LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6B</td>
<td>Everyday Computing</td>
<td>Social Networking and Communication</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6C</td>
<td>Search, Wikis, Commerce, and News</td>
<td>Cloud Computing, The Digital Divide</td>
<td>EU 7.1, EU 7.4</td>
<td>LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6D</td>
<td>Unit Project</td>
<td>Future Technology Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 1.1.1, LO 1.2.4, LO 5.1.1, LO 5.1.3, LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6G</td>
<td>Big Picture</td>
<td>Silos and Walled Gardens</td>
<td>EU 7.1, EU 7.4</td>
<td>LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6H</td>
<td>The Internet</td>
<td>Network Infrastructure Communication Protocols World Wide Web</td>
<td>EU 6.1, EU 6.2</td>
<td>LO 6.1.1, LO 6.2.1, LO 6.2.2</td>
</tr>
<tr>
<td>6K</td>
<td>Unit Project</td>
<td>Future Technology Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 1.1.1, LO 1.2.4, LO 5.1.1, LO 5.1.3, LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6L</td>
<td>Big Picture</td>
<td>Net Neutrality</td>
<td>EU 7.1, EU 7.4</td>
<td>LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6M</td>
<td>Innovations in Computing</td>
<td>Pioneers in Computing Distributed Computing Ethics of Autonomous Technology</td>
<td>EU 7.1, EU 7.4</td>
<td>LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6N</td>
<td>Coding Skills</td>
<td>Inventing the Future</td>
<td>EU 7.1, EU 7.4</td>
<td>LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6Q</td>
<td>Unit Project</td>
<td>Future Technology Project (cont.)</td>
<td>EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 1.1.1, LO 1.2.4, LO 5.1.1, LO 5.1.3, LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6R</td>
<td>Unit Exam</td>
<td>Unit 6 Exam</td>
<td>EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
<tr>
<td>6T</td>
<td>Unit Project</td>
<td>Project Presentations</td>
<td>EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4</td>
<td>LO 1.1.1, LO 1.2.4, LO 5.1.1, LO 5.1.3, LO 6.1.1, LO 6.2.1, LO 6.2.2, LO 7.1.1, LO 7.4.1</td>
</tr>
</tbody>
</table>
Unit 6: Innovative Technologies

Unit Description

As a way of further expanding upon the applications of computer science in the advancement of computational technologies, this unit aims to broaden students' awareness of the computing tools they use and rely on every day and to encourage them to start thinking about the decisions and processes that go into the creation of these technologies.

Students will begin by exploring many of the key roles that technology plays in their lives, including social networking, online communication, search, commerce, and news and examining the ways these ever-evolving technologies have impacted individuals and societies in recent years. With so many of these technologies relying on the Internet to connect users and data across varied and remote locations, the students will then "take a peek under the hood" to examine the systems and protocols that make up the global infrastructure of the Internet. Finally, students will turn their attention to the past, present, and future of computing to begin imagining the technology that might exist in their future and the role that they might play in bringing it about.

Unit Coverage

- Big Ideas: BI 1, BI 5, BI 6, BI 7
- Enduring Understandings: EU 1.1, EU 1.2, EU 5.1, EU 6.1, EU 6.2, EU 7.1, EU 7.4
- Learning Objective: LO 1.1.1 [P2], LO 1.2.4 [P6], LO 5.1.1 [P2], LO 5.1.3 [P6], LO 6.1.1 [P3], LO 6.2.1 [P5], LO 6.2.2 [P4], LO 7.1.1 [P4], LO 7.4.1 [P1]
- Computational Thinking Practices: P1, P2, P3, P4, P5, P6

Unit Topics [CR1a] [CR1c] [CR1d] [CR1e] [CR2f] [CR2g]

- Everyday Computing [EU 7.1, EU 7.4] [LO 7.1.1 [P4], LO 7.4.1 [P1]]
  - Students will explore the ways that innovations in digital technology can impact the lives of individuals and communities.
  - Students will analyze the role that digital technology plays in their everyday lives.
  - Students will analyze the role that digital technology plays in their social communications and interactions.
  - Students will explore the impact that instant access to global search, news, and information has had on individuals and communities.
  - Students will analyze the benefits and risks of cloud computing.
  - Students will investigate the socioeconomic causes and effects related to the digital divide.

- The Internet [EU 6.1, EU 6.2] [LO 6.1.1 [P3], LO 6.2.1 [P5], LO 6.2.2 [P4]]
  - Students will examine the overall design and architecture of the Internet.
  - Students will explore the role of servers, routers, gateways, and clients.
  - Students will examine the domain name system and its role in network routing.
  - Students will examine a number of standard network protocols, including IP, TCP, UDP, SMTP, HTTP, and FTP.
Unit 6: Innovative Technologies

APPLICATION

○ Students will investigate the series of components and events that are involved in the transmission of an email or SMS text over the network.
○ Students will investigate the series of components and events that are involved in the transmission of an HTML request from a Web browser.
○ Students will analyze the impact of hyperlinked documents on how individuals find, acquire, and learn new information.
○ Students will analyze the legal, social, and commercial impact that the World Wide Web has had on society.

● Innovations in Computing [EU 7.1, EU 7.4] [LO 7.1.1 [P4], LO 7.4.1 [P1]]
  ○ Students will investigate a number of key individuals and breakthroughs in the development of modern computing.
  ○ Students will explore the design goals and technological advances in the development of the modern computer.
  ○ Students will explore the design goals and technological advances in the development of the Internet.
  ○ Students will explore the design goals and technological advances in the development of human-computer interfaces.
  ○ Students will examine the roles and applications of distributed computing.
  ○ Students will examine the ethical implications of autonomous technology.
  ○ Students will investigate and extrapolate from recent advances in computing to make predictions about the capabilities of future technologies.
  ○ Students will analyze how future technologies might impact individuals and societies.

● Coding Skills [EU 7.1, EU 7.4] [LO 7.1.1 [P4], LO 7.4.1 [P1]]
  ○ Students will identify shortcomings of existing technologies.
  ○ Students will identify how individuals’ lives may be enhanced through technological innovations.
  ○ Students will develop design specifications for hypothetical, future technologies.

● Big Picture [EU 7.4] [LO 7.4.1 [P1]]
  ○ Students will discuss the benefits and risks of open versus closed platforms.

Unit Project [CR1a] [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2a] [CR2e] [CR2f] [CR2g]

● Future Technology Project
  ○ Students will collaborate in pairs to envision and design a future innovation in technology. [EU 1.1, EU 1.2, EU 5.1] [LO 1.1.1 [P2], LO 1.2.4 [P6], LO 5.1.3 [P6]]
  ○ Students will discuss and identify a specific purpose that their innovation will serve (e.g., entertainment, problem solving, education, artistic expression, etc.) and its key features. [EU 5.1] [LO 5.1.1 [P2]]
  ○ Students will evaluate the potential benefits and risks of their innovation. [EU 7.1, EU 7.4] [LO 7.1.1 [P4], LO 7.4.1 [P1]]
  ○ Students will identify existing technological resources that their innovation may utilize. [EU 6.2] [LO 6.2.2 [P4]]
Unit 6: Innovative Technologies

APPLICATION

- Students will identify technological challenges that must be overcome before their innovation can be fully realized. [EU 6.1, EU 6.2] [LO 6.1.1 [P3], LO 6.2.1 [P5], LO 6.2.2 [P4]]
- Students will develop a mock-up of their innovation that demonstrates its use and functionality. [EU 1.2] [LO 1.2.4 [P6]]
- Students will write a detailed product description and deliver an elevator pitch to the class detailing the features of their innovation and its potential impact on society using appropriate terminology. [LO 5.1.3 [P6]]
- Students will provide written feedback to their peers on the potential of each collaborative team's design. [LO 5.1.3 [P6]]

Unit Readings

- *Blown to Bits* (Abelson, Ledeen, Lewis). Appendix – The Internet as System and Spirit

Unit Assessments

- Minor exercises addressing specific unit topics and objectives
- Formally assessed, multiple-choice test addressing unit objectives (single- and multiple-select questions)
- Rubric-assessed, individual and/or collaborative unit project demonstrating mastery of unit objectives

Unit Objectives

- Big Idea 1: Creativity
  - Apply an iterative and exploratory development process to create a computational artifact using non-prescribed techniques, novel combinations of artifacts, and/or personal curiosities. [EK 1.1.1A, EK 1.1.1B] [P2]
  - Use appropriate collaboration tools and techniques to create a computational artifact. [EK 1.2.4A, EK 1.2.4B] [P6]
  - Use appropriate interpersonal skills, communication, and group decision-making to create an enhanced, collaborative computational artifact. [EK 1.2.4C, EK 1.2.4D] [P6]
  - Create a collaborative computational artifact that reflects the diverse talents and personal ideas of all group members. [EK 1.2.4E, EK 1.2.4F] [P6]
- Big Idea 5: Programming
  - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge using visual, audible, or tactile inputs and outputs. [EK 5.1.1B] [P2]
  - Identify additional desired outcomes for a program that extend beyond the original purpose of a program. [EK 5.1.1D] [P2]
Unit 6: Innovative Technologies

○ Explain how 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. [EK 5.1.1E] [P2]
○ Use collaboration to facilitate multiple perspectives in developing ideas for solving problems by programming. [EK 5.1.3B] [P6]
○ Use effective communication between participants in the iterative development of a program. [EK 5.1.3C, EK 5.1.3F] [P6]

● Big Idea 6: The Internet
○ Explain how world-wide collaboration is enabled through the end-to-end architecture that consists of unique addresses and standard protocols for connecting new devices and networks on the Internet. [EK 6.1.1A, EK 6.1.1B, EK 6.1.1C, EK 6.1.1D] [P3]
○ Explain how the domain name system (DNS) translates names to IP addresses that are assigned to every device connected to the Internet. [EK 6.1.1E, EK 6.1.1G] [P3]
○ Explain the role of evolving Internet standards and its relation to the need for a new Internet protocol (IPv6). [EK 6.1.1F, EK 6.1.1H] [P3]
○ Explain how the Internet Engineering Task Force (IETF) establishes and oversees key Internet standards, such as hypertext transfer protocol (HTTP), Internet protocol (IP), and simple mail transfer protocol (SMTP). [EK 6.1.1I] [P3]
○ Explain how the hierarchical design of the Internet's routing and addressing systems (domain name syntax, IP addresses) provide fault tolerance and redundancy. [EK 6.2.1A, EK 6.2.1B, EK 6.2.1C, EK 6.2.1D] [P5]
○ Explain how the hierarchy and redundancy of routing with the domain name system (DNS) help the Internet to scale to more devices and more people. [EK 6.2.2A, EK 6.2.2B, EK 6.2.2C] [P4]
○ Explain how open standards and well-specified interfaces and protocols enable widespread growth and use of the Internet. [EK 6.2.2D, EK 6.2.2E] [P4]
○ Explain the importance of standards in sharing and transmitting data and control information through a packet-switched system, such as transmission control protocol/Internet protocol (TCP/IP), hypertext transfer protocol (HTTP), and secure sockets layer/transport layer security (SSL.TLS). [EK 6.2.2F, EK 6.2.2G, EK 6.2.2H] [P4]
○ Explain how the bandwidth and latency of a system affect its use. [EK 6.2.2I, EK 6.2.2J, EK 6.2.2K] [P4]

● Big Idea 7: Global Impact
○ Analyze the impact and use of today's communication technologies on their lives (including email, SMS, chat, video conferencing, social media, etc.). [EK 7.1.1A, EK 7.1.1B, EK 7.1.1C, EK 7.1.1H] [P4]
○ Analyze the advantages and solutions enabled by cloud computing and instant access to public data (e.g., search engines, wikis, product reviews, etc.). [EK 7.1.1D, EK 7.1.1E, EK 7.1.1F] [P4]
Unit 6: Innovative Technologies

APPLICATION

- Identify new technologies and applications made possible by the proliferation of inexpensive sensors and processors. [EK 7.1.G] [P4]
- Analyze how sensor-driven technology (e.g., GPS, sensor networks, smart grids/building/transportation, assistive technologies, etc.) have changed human behavior and enhanced human capabilities. [EK 7.1.I, EK 7.1.J, EK 7.1.K, EK 7.1.L] [P4]
- Analyze the impact and use of the Internet and the Web for communication, e-commerce, healthcare, entertainment, and online learning. [EK 7.1.M, EK 7.1.N] [P4]
- Identify the positive and negative effects of the Internet and the Web on productivity. [EK 7.1.O] [P4]
- Explain how social media, online access, and the "digital divide" affect individuals and socioeconomic groups differently around the world. [EK 7.4.A, EK 7.4.D] [P1]
- Explain how mobile, wireless, and networked computing impact innovation throughout the world. [EK 7.4.B] [P1]
- Explain how the global distribution of computing resources raises issues of equity, access, and power. [EK 7.4.C] [P1]
- Explain commercial and governmental initiatives support networks and infrastructure. [EK 7.4.E] [P1]
## Unit 7: Performance Tasks

### Unit Schedule

<table>
<thead>
<tr>
<th>Day</th>
<th>Topic</th>
<th>Lessons</th>
<th>Enduring Understandings</th>
<th>Learning Objectives</th>
</tr>
</thead>
<tbody>
<tr>
<td>7A</td>
<td>Performance Tasks</td>
<td>Performance Tasks Introduction</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ET</td>
<td>Explore – Impact of Computing Innovations</td>
<td>Identify Topic of Research</td>
<td>EU 1.2, EU 3.3, EU 7.1, EU 7.2, EU 7.3, EU 7.4</td>
<td>LO 1.2.5, LO 3.3.1, LO 7.1.1, LO 7.2.1, LO 7.3.1, LO 7.4.1</td>
</tr>
<tr>
<td></td>
<td>8 hours [CR3] (Approximately 2 weeks)</td>
<td>Conduct Independent Research</td>
<td>EU 1.2, EU 7.1, EU 7.2, EU 7.3, EU 7.4</td>
<td>EU 1.2.1, EU 1.2.2, EU 1.2.3, EU 7.1.1, EU 7.2.1, EU 7.3.1, EU 7.4.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Responses to Prompts</td>
<td>EU 5.2, EU 5.4</td>
<td>LO 5.2.1, LO 5.4.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Create Computational Artifact</td>
<td>EU 1.2, EU 7.1, EU 7.2, EU 7.3, EU 7.4</td>
<td>EU 1.2.2, EU 1.2.3, EU 1.2.5, EU 7.1.1, EU 7.2.1, EU 7.3.1, EU 7.4.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Support for Computational Artifact</td>
<td>EU 1.2, EU 7.1, EU 7.2, EU 7.3, EU 7.4</td>
<td>LO 1.2.2, LO 1.2.3, LO 1.2.5, LO 7.1.1, LO 7.2.1, LO 7.3.1, LO 7.4.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Submit &quot;Explore&quot; Task Computational Artifact and Written Responses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CT</td>
<td>Create – Applications from Ideas</td>
<td>Identify Project Ideas</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>12 hours [CR4] (Approximately 3 weeks)</td>
<td>Develop, Implement, and Test Program</td>
<td>EU 1.2, EU 2.2, EU 4.1, EU 5.1, EU 5.3, EU 5.4, EU 5.5</td>
<td>LO 1.2.1, LO 1.2.2, LO 1.2.3, LO 1.2.4, LO 2.2.1, LO 2.2.2, LO 4.1.2, LO 5.1.1, LO 5.1.2, LO 5.1.3, LO 5.3.1, LO 5.4.1, LO 5.5.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Create Video of Program</td>
<td>EU 1.2</td>
<td>LO 1.2.1, LO 1.2.2, LO 1.2.3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Write Responses on Program</td>
<td>EU 1.2, EU 5.2, EU 5.3</td>
<td>LO 1.2.1, LO 1.2.2, LO 1.2.3, LO 5.2.1, LO 5.3.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Submit &quot;Create&quot; Task Program, Video, and Written Responses</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Unit Description

This unit serves to fulfill the Performance Task requirements of the AP Computer Science Principles exam. This externally moderated assessment will account for 40% of the student's AP exam score. As such, the work produced in this unit should reflect the sole work of the student and performed in-class with minimal involvement from the classroom teacher. For the "Create"
Unit 7: Performance Tasks

Performance Task, the student may receive collaborative support from a fellow student, but the work submitted should the individual student's own work.

By this point in the course, all of the projects, exercises, and classroom discussions from the previous six units will have provided students with extensive, hands-on experience with the exploration, use, and creation of computational artifacts in a variety of contexts. In this unit, students will draw upon those collective skills to demonstrate mastery of essential course concepts by completing the "Explore" and "Create" Performance Tasks that make up the AP through-course assessment.

For the "Explore – Impact of Computing Innovations" Performance Task, students will demonstrate their ability to conduct independent research into an innovative technology and intelligently discuss its impact and influence on society as a whole. And for the "Create – Applications from Ideas" Performance Task, students will demonstrate their ability to work individually and collaboratively to design and develop a functional program for solving a problem and/or self-expression.

Unit Coverage

- Big Ideas: BI 1, BI 2, BI 3, BI 4, BI 5, BI 7
- Enduring Understandings: EU 1.2, EU 2.2, EU 3.3, EU 4.1, EU 5.1, EU 5.2, EU 5.3, EU 5.4, EU 5.5, EU 7.1, EU 7.2, EU 7.3, EU 7.4
- Learning Objectives: LO 1.2.1 [P2], LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 1.2.5 [P4], LO 2.2.1 [P2], LO 2.2.2 [P3], LO 3.3.1 [P4], LO 4.1.1 [P2], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3], LO 5.3.1 [P3], LO 5.4.1 [P4], LO 5.5.1 [P1], LO 7.1.1 [P4], LO 7.2.1 [P1], LO 7.3.1 [P4], LO 7.4.1 [P1]
- Computational Thinking Practices: P1, P2, P3, P4, P6

Unit Topics [CR1a] [CR1b] [CR1c] [CR1d] [CR1e] [CR1f] [CR2a] [CR2b] [CR2c] [CR2d] [CR2e] [CR2g]

- Exploration and Research [EU 1.2, EU 3.3, EU 7.1, EU 7.2, EU 7.3, EU 7.4, EU 7.5] [LO 1.2.1 [P2], LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.5 [P4], LO 3.3.1 [P4], LO 7.1.1 [P1], LO 7.3.1 [P4], LO 7.4.1 [P1], LO 7.5.1 [P1], LO 7.5.2 [P5]]
  - Students will conduct independent research into a technological innovation of their choice.
  - Students will examine the social, economic, and cultural impact of their chosen technological innovation.
  - Students will examine how their chosen technological innovation consumes, produces, and/or transforms data.
  - Students will identify and discuss concerns about data storage, data privacy, or data security with regard to their chosen technological innovation.
- Creative Development [EU 1.2, 2.2, EU 4.1, EU 5.1, EU 5.2, EU 5.3, EU 5.4, EU 5.5, EU 7.5] [LO 1.2.1 [P2]], LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 2.2.1 [P2], LO 2.2.2 [P3], LO
Unit 7: Performance Tasks

4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3], LO 5.3.1 [P3], LO 5.4.1 [P4], LO 5.5.1 [P1], LO 7.5.1 [P1], LO 7.5.2 [P5]

- Students will individually and/or collaboratively design, implement, and test a program designed to solve a problem of interest to them.
- Students will document the functionality of their program and reflect on its development process.

Unit Project [CR3] [CR4]

- "Explore – Impact of Computing Innovations" Performance Task [EU 1.2, EU 3.3, EU 7.1, EU 7.2, EU 7.3, EU 7.4, EU 7.5] [LO 1.2.1 [P2], LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.5 [P4], LO 3.3.1 [P4], LO 7.1.1 [P4], LO 7.2.1 [P1], LO 7.3.1 [P4], LO 7.4.1 [P1], LO 7.5.1 [P1], LO 7.5.2 [P5]]
  - This project will encompass 8 hours of in-class, independent research and work.
  - Each student will investigate a computing innovation of his/her choice that has had a significant impact on society, economy, or culture.
  - Each student will produce a computational artifact that describes the intended purpose and function of the computing innovation and demonstrates how it fulfills that purpose.
  - Each student will document the development process, tools, and techniques used in creating the computational artifact.
  - Each student will identify and explain the beneficial and harmful effects of the computing innovation on society, economy, or culture.
  - Each student will identify and discuss how the computing innovation consumes, produces, and/or transforms data and address concerns about data storage, data privacy, or data security with regard to their chosen technological innovation.
  - Each student will thoroughly cite the sources used in conducting their research of the computing innovation.
  - The product of this project, including the computational artifact and written responses, will serve as part of the student's formal submission to the College Board for the AP Computer Science Principles exam.

- "Create – Applications from Ideas" Performance Task [EU 1.2, EU 2.2, EU 4.1, EU 5.1, EU 5.2, EU 5.3, EU 5.4, EU 5.5, EU 7.5] [LO 1.2.1 [P2], LO 1.2.2 [P2], LO 1.2.3 [P2], LO 1.2.4 [P6], LO 2.2.1 [P2], LO 2.2.2 [P3], LO 4.1.2 [P5], LO 5.1.1 [P2], LO 5.1.2 [P2], LO 5.1.3 [P6], LO 5.2.1 [P3], LO 5.3.1 [P3], LO 5.4.1 [P4], LO 5.5.1 [P1], LO 7.5.1 [P1], LO 7.5.2 [P5]]
  - This project will encompass 12 hours of in-class, independent and/or collaborative work.
  - Each student will design, implement, and test a program that solves a problem of personal interest to the student.
  - Each student will describe and reflect on their role in the development of the program.
Unit 7: Performance Tasks

○ Students will make a 1-minute video demonstrating the use and functionality of the program.
○ Students may work collaboratively on their project, but each student will be solely responsible for developing at least one significant part of their program.
○ The product of this project, including the program, video, and written responses, will serve as part of the student's formal submission to the College Board for the AP Computer Science Principles exam.

Unit Readings

● Miscellaneous, student-selected resources (varies by student)

Unit Assessments

● A student-produced computational artifact with written responses to questions about the artifact and computing innovation
● An individually or collaboratively produced program with documenting video and written response detailing the development process

Unit Objectives

● Big Idea 1: Creativity
  ○ Design and create a computational artifact (e.g., program, image, audio, video, presentation, etc.) for creative expression using appropriate software tools and techniques (e.g., programming IDEs, spreadsheet, 3D printer, text editor, etc.). [EK 1.2.1A, EK 1.2.1B, EK 1.2.1C, EK 1.2.1D, EK 1.2.1E] [P2]
  ○ Create a computational artifact using computing tools and innovative, non-traditional techniques to solve a problem. [EK 1.2.2A, EK 1.2.2B] [P2]
  ○ Create a computational artifact by combining and modifying existing artifacts to show personal expression of ideas. [EK 1.2.3A, EK 1.2.3C] [P2]
  ○ Use computational tools to create or modify a computational artifact with enhanced detail and precision. [EK 1.2.3B] [P2]
  ○ Use appropriate collaboration tools and techniques to create a computational artifact. [EK 1.2.4A, EK 1.2.4B] [P6]
  ○ Use appropriate interpersonal skills, communication, and group decision-making to create an enhanced, collaborative computational artifact. [EK 1.2.4C, EK 1.2.4D] [P6]
  ○ Create a collaborative computational artifact that reflects the diverse talents and personal ideas of all group members. [EK 1.2.4E, EK 1.2.4F] [P6]
  ○ Analyze the correctness, usability, functionality, and suitability of a computational artifact in terms of the context in which it is used or perceived. [EK 1.2.5A, EK 1.2.5C, EK 1.2.5D] [P4]
  ○ Analyze a computational artifact for weaknesses, mistakes, and errors. [EK 1.2.5B] [P4]
● Big Idea 2: Abstraction
Unit 7: Performance Tasks

- Develop an abstraction by identifying common features and removing detail in order to generalize concepts and functionality. [EK 2.2.1A, EK 2.2.1B]
- Develop software using multiple levels of abstraction, including constants, expressions, statements, procedures, and libraries, to more effectively apply available resources and tools to solve problems. [EK 2.2.2A, EK 2.2.2B]

- Big Idea 3: Data and Information
  - Analyze how the characteristics of data, the methods and costs of manipulating the data, and the intended uses of data relate to the storage requirements and choice of storage media. [EK 3.3.1G, EK 3.3.1H, EK 3.3.1I] [P4]

- Big Idea 4: Algorithms
  - Develop an algorithm using sequencing, selection, and iteration. [EK 4.1.1A, EK 4.1.1B, EK 4.1.1C, EK 4.1.1D] [P2]
  - Develop an algorithm that uses or combines existing, standard algorithms to ensure correctness of the resulting solution. [EK 4.1.1E, EK 4.1.1F, EK 4.1.1G] [P2]
  - Express algorithms in natural language and pseudocode for human readability. [EK 4.1.2B]
  - Express algorithms in a programming language for execution by a computer. [EK 4.1.2C]
  - Construct algorithms using sequencing, selection, and iteration. [EK 4.1.2G]

- Big Idea 5: Programming
  - Develop a variety of programs using methods and techniques that are appropriate for the goals of the programmer. [EK 5.1.1A] [P2]
  - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge using visual, audible, or tactile inputs and outputs. [EK 5.1.1B] [P2]
  - Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge using standards or methods that differ from those used for programs developed for widespread distribution. [EK 5.1.1C] [P2]
  - Develop a large, correct program using an iterative process that incrementally combines tested program components. [EK 5.1.2A, EK 5.1.2B, EK 5.1.2C] [P2]
  - Provide documentation about program components, such as blocks and procedures, to maintain correct programs when working individually or collaboratively with other programmers. [EK 5.1.2D, EK 5.1.2E, EK 5.1.2F] [P2]
  - Consult and communicate with program users to identify concerns that affect the solution to problems. [EK 5.1.2G, EK 5.1.2H] [P2]
  - Develop a program using appropriate knowledge and skill of the development process, including designing, implementing, testing, debugging, and maintaining programs. [EK 5.1.2I, EK 5.1.2J] [P2]
  - Use collaboration to decrease the size and complexity of tasks required of individual programmers and to develop program components independently. [EK 5.1.3A, EK 5.1.3E] [P6]
○ Use collaboration to facilitate multiple perspectives in developing ideas for solving problems by programming. [EK 5.1.3B] [P6]
○ Use effective communication between participants in the iterative development of a program. [EK 5.1.3C, EK 5.1.3F] [P6]
○ Use collaboration to find and correct errors with developing programs. [EK 5.1.3D] [P6]
○ Explain how algorithms are implemented using program instructions that are processed sequentially during program execution. [EK 5.2.1A, EK 5.2.1B, EK 5.2.1D] [P3]
○ Explain how program instructions may involve variables that are initialized and updated, read, and written. [EK 5.2.1C] [P3]
○ Explain how executable programs and automation increase the scale of problems and sets of problems that can be addressed. [EK 5.2.1I, EK 5.2.1J] [P3]
○ Use abstraction to create named, parameterized, and reusable blocks of programming in order to reduce the complexity of writing and maintaining a program. [EK 5.3.1A, EK 5.3.1B, EK 5.3.1C, EK 5.3.1D] [P3]
○ Use parameterization to generalize specific solutions and allow a single function to be used in place of duplicated code. [EK 5.3.1E, EK 5.3.1F, EK 5.3.1G] [P3]
○ Employ data abstraction and its ability to separate behavior from implementation by using a variety of abstract data types, including strings, integers, floating-point numbers, and lists. [EK 5.3.1H, EK 5.3.1I, EK 5.3.1J, EK 5.3.1K] [P3]
○ Use lists and procedures as abstractions in programming to produce programs that are easier to develop and maintain. [EK 5.3.1L] [P3]
○ Use well-documented application program interfaces (APIs) and libraries to connect software components and to simplify complex programming. [EK 5.3.1M, EK 5.3.1N, EK 5.3.1O] [P3]
○ Use good programming style, such as meaningful names for variables and procedures, shorter code blocks, and non-duplicated code, in order to improve the determination of program correctness. [EK 5.4.1A, EK 5.4.1B, EK 5.4.1C, EK 5.4.1D] [P4]
○ Debug a program by locating and correcting errors. [EK 5.4.1E] [P4]
○ Describe the functionality of a program at a high level in terms of what it does and how a user interacts with it and provide examples of intended behavior on specific inputs in order to find program errors. [EK 5.4.1F, EK 5.4.1G, EK 5.4.1L, EK 5.4.1M, EK 5.4.1N] [P4]
○ Use visual displays (or different modalities) of program state to help in finding errors. [EK 5.4.1H] [P4]
○ Construct mathematical and logical expressions using arithmetic and Boolean operators. [EK 5.5.1D, EK 5.5.1E, EK 5.5.1F] [P1]
○ Employ intuitive and formal reasoning about program components using Boolean concepts. [EK 5.5.1G] [P1]
Unit 7: Performance Tasks

- Employ lists and collections as abstract data types (ADTs) that provide functionality to add, remove, and iterate over all elements, as well as to determine whether an element is in a collection. [EK 5.5.1H, EK 5.5.1I, EK 5.5.1J] [P1]

- Big Idea 7: Global Impact
  - Explain how machine learning and data mining have enabled innovations in medicine, business, and science. [EK 7.2.1A] [P1]
  - Explain how computing enables innovation and creativity in scientific and other fields. [EK 7.2.1B, EK 7.2.1C, EK 7.2.1G] [P1]
  - Analyze the legal and ethical concerns of open source and licensed software, libraries, and code. [EK 7.3.1F, EK 7.3.1Q] [P4]
  - Analyze the intellectual property and copyright concerns with digital information, audio, video, and textual content. [EK 7.3.1N, EK 7.3.1O, EK 7.3.1P] [P4]
  - Explain how social media, online access, and the "digital divide" affect individuals and socioeconomic groups differently around the world. [EK 7.4.1A, EK 7.4.1D] [P1]
  - Use online databases and libraries to access information. [EK 7.5.1A] [P1]
  - Use advanced search tools, Boolean logic, and key words to focus or limit searches to desired results. [EK 7.5.1B] [P1]
  - Avoid plagiarism by appropriately acknowledging sources. [EK 7.5.1C] [P1]
  - Evaluate the credibility and relevance of sources of information. [EK 7.5.2A, EK 7.5.2B] [P5]