Computer Science Principles	Scope and Sequence
Unit Lesson	**Objectives**
COMPUTING SYSTEMS	
Meet Your Operating System	
	Describe the role of an operating system.
How Good Is Your Memory?	
	Describe the role of memory.
Finding Where You Put Things	
	Describe how data are stored and retrieved.
Who Gets Attention?	
	Describe how multiple, concurrent processes are managed.
Authorized Processes Only	
	Describe how processes are authorized.
Computer System Criteria	
	Describe criteria for purchasing or upgrading computer system hardware (e.g., Wi-Fi, mobile devices, home and office machines).
Project: Buying a Computer	
	Evaluate criteria for purchasing or upgrading computer system hardware (e.g., Wi-Fi, mobile devices, home and office machines).
When Things Go Wrong	
	Describe common problems and their solutions.
My Motherboard Is Smarter than Your Motherboard	
	Describe the function of the CPU and RAM.
What's My Bit?	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Perform conversions between binary representation of numeric and nonnumeric data.
	You Will Never Walk through This Gate	
		Describe how a logic gate works.
	Input and Output	
		Differentiate between input, output, and I/O pins.
	Logic Circuits	
		Identify the output for given logic circuits.
	Project: Logic Circuits	
		Draw a logic circuit with three inputs and at least five gates.
		Describe the output given all possible combinations of inputs.
	Review: Computing Systems	
	Test: Computing Systems	
NETWORKS AND THE INTERNET		
	How Networks Function	
		Describe the physical parts of a network and how it functions.
	How Does Data Travel?	
		Describe how data are transmitted through a network.
	My Network Is Busy	
		Describe the issues that impact network functionality (bandwidth and load).
	My Network is Not Working	
		Describe issues that impact network functionality (topology, hardware).
	Project: Network Problem	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Solving	
		Design two scenarios, each containing a problematic network.
		Explain the problems within each network.
		Describe the steps needed to resolve the problems.
	Security Concerns	
		Identify security concerns.
	Solutions to Security Concerns	
		Identify solutions to security concerns.
	Past and Present Scams	
		Describe recent examples of security attacks using scams.
	Authentication Strategies	
		Identify practices that improve computer security and protect devices and information from unauthorized access using authentication strategies.
	Encryption and Safeguarding Keys	
		Describe best practices that improve computer security and protect devices and information from unauthorized access using encryption.
	How Processes Share Computer Resources	
		Describe threads of execution.
	Project: Cybersecurity Policy	
		Recommend a computer security policy.
	Review: Networks and the Internet	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Test: Networks and the Internet	
DATA AND ANALYSIS		
	Who's Watching You?	
		Describe ways personal data are collected from internet users.
	Who's Watching Your Phone?	
		Describe security concerns and solutions related to cell phones.
	Who's Watching Your Car or Truck?	
		Describe security concerns and solutions related to vehicles.
	Data Pattern Case Studies	
		Interpret results of data tracking.
	Picturing Data	
		Interpret visual representations of data.
	Collecting Data	
		Describe data collection techniques.
	Project: Compare Data Tools	
		Compare data collection tools, giving examples of beneficial and dishonest uses of data tracking.
	Data Analysis Software	
		Describe common software used to analyze data.
	Mathematical Calculations	
		Interpret mathematical expressions, including addition, subtraction, multiplication, real division, integer division, and modulus division.
	Your First Program	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Interpret programs that manipulate numbers and strings.
	Loops	
		Interpret loops.
	Plan Your Program	
		Use flowcharts and pseudocode to plan and document a program.
	Using the Random and Math Libraries	
		Use random and math functions (abs, square root, round).
	Project: Data Analysis	
		Implement an algorithm that analyzes data.
	Review: Data and Analysis	
	Test: Data and Analysis	
AI AND ALGORITHMS		
	Artificial Intelligence: Then and Now	
		Describe artificial intelligence.
	AI Software	
		Describe how AI is used in various types of software.
	AI in Physical Systems	
		Identify how AI is used to control physical systems.
	An AI Decision Tree	
		Interpret decision trees to create an AI algorithm.
	Slice of Machine Learning	

Unit	Lesson	Objectives
		Describe characteristics of machine learning.
	Project: A Decision Tree	
		Plan an artificial intelligence algorithm using a decision tree.
		Create a decision tree that helps individuals select which mobile phones to buy, based on at least two criteria.
	Decision Making	
		Use simple boolean conditions to make decisions.
	More about Decision Making	
		Use compound boolean conditions to make decisions.
	Searching	
		Use a loop for searching.
	Reusing Code with Functions	
		Interpret functions, both with and without parameters.
	Sorting and Efficiency	
		Interpret sorting algorithms.
	Evaluating Algorithms	
		Evaluate algorithms in terms of their efficiency, correctness, and clarity.
	Choosing Control Structures	
		Compare while loops to other control structures.
	Project: Solve Problems with Control Structures	
		Use control structures (loops and decision-making) and functions to solve a practical problem.
	Review: AI and Algorithms	
	Test: AI and Algorithms	

Computer Science Principles	Scope and Sequence
Unit Lesson	**Objectives**
VARIABLES AND CONTROL	
Organizing Data in Your Program	
	Interpret lists.
Multidimensional Lists	
	Interpret multidimensional lists.
Dictionaries	
	Compare and contrast data structures.
Other Searching Algorithms	
	Describe algorithms used for searching.
Sorting and Searching Data Structures	
	Describe how to sort and search a list.
Pros and Cons for Data Structure Use	
	Compare scenarios to the type of data structure used.
Project: Solving Problems with Data Structures	
	Construct a solution to a practical problem using a one-dimensional list or other data structure.
Recursion	
	Describe recursion.
Comparing Algorithms	
	Compare recursion to other methods.
Intractable Problems	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Describe intractable problems.
	Project: Recursive Algorithms	
		Implement a recursive algorithm.
	Review: Variables and Control	
	Test: Variables and Control	
SEMESTER 1 REVIEW AND EXAM		
	Semester 1 Review: Computer Science Principles	
	Semester 1 Exam: Computer Science Principles	
MODULARITY		
	You Design the Data Structure	
		Interpret user-defined data structures.
	Using the Data Structure You Design	
		Interpret applications of user-defined data structures.
	Creating a Class for Your Data Structure	
		Use classes to create and manipulate objects.
	Comparing Objects	
		Compare two objects.
	Project: Solving Problems with Modularity	
		Construct a solution to a problem using data structures, classes, and functions, working in a group.
	Large-Scale Computational	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Problems	
		Identify characteristics of large-scale computational problems.
	Decompose Problems into Manageable Subproblems	
		Identify generalizable patterns in a large-scale computation problem that can be applied to a solution.
	Finding Solutions to Large-Scale Problems	
		Describe how to find solutions to large-scale computational problems.
	Using Libraries	
		Describe how libraries can be used to solve programming problems.
	Creating Web Pages	
		Use and interpret basic HTML.
	Using APIs	
		Describe how to use application programming interfaces (APIs).
	Project: Identifying Patterns	
		Demonstrate code reuse by creating a solution using libraries to identify generalizable patterns in large data sets with real-world applications.
		Use peer review and pseudocode documentation.
	Review: Modularity	
	Test: Modularity	
DEVELOPMENT AND SECURITY		
	What Is a Software Development Life Cycle Process?	
		Describe the software development life cycle.

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Examples of the SDLC	
		Identify steps to implement the software development life cycle.
	Designing Software for Everyone	
		Describe how to use the software development life cycle to make programs accessible to all.
	Project: Stages of SDLC	
		Plan a solution incorporating each stage of the software development life cycle.
	The Test Stage of the SDLC	
		Identify how to test software.
	Debugging	
		Identify types of bugs and techniques to resolve them.
	Deployment of the SDLC	
		Describe the deployment stage of the software development life cycle.
	Project: Working as a Team	
		Use the software development life cycle process to plan and refine a solution, working as a team.
	Security Issues Then and Now	
		Describe security issues that arise when programming.
	Security of Web Programming	
		Describe security threats to web-based programs.
	Security Solutions	
		Describe solutions to programming security vulnerabilities.
	Safeguards in Programs	
		Describe how a program can implement precautions to avoid invalid data.

Unit	Lesson	Objectives
	Project: Security Issues	
		Refine code that has vulnerabilities.
	Review: Development and Security	
	Test: Development and Security	
PLATFORMS AND DEVELOPMENT TOOLS		
	Desktop, Web, and Mobile Development	
		Compare and contrast multiple computing platforms.
	Web Development	
		Describe techniques of web development.
	Mobile Apps	
		Describe techniques of mobile development.
	Can One App Do It All?	
		Describe issues when writing a single application used on multiple platforms.
	Project: Plan a Mobile App	
		Develop a plan for a mobile app.
	Version Control Systems	
		Describe version control systems.
	Integrated Development Environments (IDEs)	
		Describe integrated development environments (IDEs).
	Collaborative Tools	

Unit	Lesson	Objectives
		Describe collaborative tools.
	Collaborative Best Practices	
		Describe best practices for collaborations.
	Collaborative Pitfalls	
		Describe problems that arise during collaborations.
	Project: Platforms and Development Tools	
		Plan, build, test, and refine a programming solution as a collaborative project, documenting versions.
	Review: Platforms and Development Tools	
	Test: Platforms and Development Tools	
ALGORITHMS AND PROGRAMMING		
	Testing Your Software	
		Identify test cases to verify a simple program functions correctly.
	Identifying Important Test Cases	
		Identify important test cases.
	Develop Test Cases	
		Identify appropriate test cases.
	Use Test Cases	
		Determine whether a program is correct using test cases.
	Project: Using Test Cases	
		Write a program that performs a calculation and assigns the results to a category, then verify its correctness with test cases.

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Create and Document Modifications	
		Identify how to document modifications.
	When the Fix Is the Problem	
		Determine when a fix causes problems.
	Code Review	
		Describe the code review process.
	Code Review Examples	
		Describe how steps of code review can be implemented.
	Other Programming Languages	
		Compare multiple programming languages.
	Best Programs for the Purpose	
		Determine how different languages' features make them suitable for solving different types of problems.
	Project: Code Review	
		Create a code review for another student's mid-unit project.
	Review: Algorithms and Programming	
	Test: Algorithms and Programming	
IMPACTS OF COMPUTING		
	Can a Program Help You?	
		Describe how computational artifacts can have beneficial effects.
	Can a Program Hurt You?	

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Describe how computational artifacts can have harmful effects.
	Reducing Negative Impacts	
		Describe ways to reduce the negative impact of a computational artifact.
	Equity and Access in a Global Society	
		Identify how infrastructure and funding impact equitable access to computing resources for underrepresented groups (e.g. race, ethnicity, gender, geographic location, socioeconomic status).
	Influence of Global Resources	
		Identify how net neutrality and government regulations impact equitable access to computing resources for underrepresented groups (e.g., race, ethnicity, gender, geographic location, socioeconomic status).
	The Revolution of Computational Innovation	
		Describe how computational innovations have revolutionized aspects of our culture.
	Project: Plan a Performance Task	
		Plan a program that will be used to meet the requirements of the course culminating performance task.
	Impacts of Computing on Business	
		Describe the impact of computing on business and commerce (e.g., automated inventory processing, financial transactions, e-commerce, virtualization, and cloud computing).
	Impacts of Privacy and Security	
		Describe privacy issues.
	Software Laws and Regulations	
		Describe software laws and regulations.

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
	Should Intellectual Property Be Shared?	
		Describe how different types of software licenses (e.g., open source and proprietary licenses) can be used to share and protect intellectual property.
	Recognizing High-Quality Resources	
		Determine the reliability of digital resources.
	How to Give Credit	
		Describe methods used to avoid plagiarism.
	Project: Bias and Equity	
		Research and analyze a computational source of bias, then propose refinements to reduce bias, increase equity, and support accessibility.
	Review: Impacts of Computing	
	Test: Impacts of Computing	
SEMESTER 2 REVIEW AND EXAM		
	Semester 2 Review: Computer Science Principles	
		Interpret user-defined data structures.
		Interpret applications of user-defined data structures.
		Use classes to create and manipulate objects.
		Compare two objects.
		Identify characteristics of large-scale computational problems.
		Identify generalizable patterns in a large-scale computation problem that can be applied to a solution.
		Describe how to find solutions to large-scale computational problems.
		Describe how libraries can be used to solve programming problems.

Computer Science Principles		Scope and Sequence
Unit	**Lesson**	**Objectives**
		Use and interpret basic HTML.
		Describe how to use application programming interfaces (APIs).
		Describe the software development life cycle.
		Identify steps to implement the software development life cycle.
		Describe how to use the software development life cycle to make programs accessible to all.
		Identify how to test software.
		Identify types of bugs and techniques to resolve them.
		Describe the deployment stage of the software development life cycle.
		Describe security issues that arise when programming.
		Describe security threats to web-based programs.
		Describe solutions to programming security vulnerabilities.
		Describe how a program can implement precautions to avoid invalid data.
	Semester 2 Exam: Computer Science Principles	
FINAL EXAM		
	Culminating Project: Computing Artifact	
		Create a plan, write a program, test it, and present the project to an audience.
	Course Review: Computer Science Principles	
	Final Exam: Computer Science Principles	