

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Course Description

Introduction to Computer Science is a full-year course that meets on a rotating basis for three (3) 55-minute blocks and one (1) 40-minute block for every five (5) day cycle.

This course is designed to offer an introduction to computer science. Students will learn the basics of computer programming along with the basics of computer science. The primary language for the course is Python. The material emphasizes computational thinking and helps develop the ability to solve complex problems. This course covers the basic building blocks of programming along with other central elements of computer science. It gives a foundation in the tools used in computer science and prepares students for further study in computer science.

Course Overview and Pacing Guide

Unit	Topic	Time Frame
1	Beginning in Computer Science	4 Weeks
2	Number Calculations and Data	4 Weeks
3	Making Decisions	3 Weeks
4	Repetition with Loops	5 Weeks
5	Programming in Earsketch	3 Weeks
6	Graphics	3 Weeks
7	Functions	3 Weeks
8	Lists	3 Weeks
9	2D Lists	3 Weeks
10	Programming in Earsketch	3 Weeks
11	Internet	4 Weeks

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Computer Science and Design Thinking (Standard 8)	
Core Idea	Performance Expectation
The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.	8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
A computing system involves interaction among the user, hardware, application software, and system software.	8.1.12.CS.2: Model interactions between application software, system software, and hardware. 8.1.12.CS.3: Compare the functions of application software, system software, and hardware.
Successful troubleshooting of complex problems involves multiple approaches including research, analysis, reflection, interaction with peers, and drawing on past experiences.	8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
The scalability and reliability of the Internet are enabled by the hierarchy and redundancy in networks. Network topology is determined by many characteristics.	8.1.12.NI.1: Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
Network security depends on a combination of hardware, software, and practices that protect data while it is at rest, in transit, and in use.	8.1.12.NI.2: Evaluate security measures to address various common security threats. 8.1.12.NI.3: Explain how the needs of users and the sensitivity of data determine the level of security implemented.
Choices individuals make about how and where data is organized and stored affects cost, speed, reliability, accessibility, privacy, and integrity.	8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored. 8.1.12.DA.3: Translate between decimal numbers and binary numbers. 8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users. 8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Career Readiness, Life Literacies, and Key Skills (Standard 9)	
Core Idea	Performance Expectation
There are strategies to improve one's professional value and marketability.	9.2.12.CAP.3: Investigate how continuing education contributes to one's career and personal growth.
Career planning requires purposeful planning based on research, self-knowledge, and informed choices.	9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills. 9.2.12.CAP.8: Determine job entrance criteria (e.g., education credentials, math/writing/reading comprehension tests, drug tests) used by employers in various industry sectors
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas
Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed.	9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice 9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving
Laws govern many aspects of computing, such as privacy, data, property, information, and identity. These laws can have beneficial and harmful effects, such as expediting or delaying advancements in computing and protecting or infringing upon people's rights.	9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics 9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users
Laws govern the use of intellectual property and there are legal consequences to utilizing or sharing another's original works without permission or appropriate credit.	9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content
Digital communities influence many aspects of society, especially the workforce. The increased connectivity between people in different cultures and different career fields have changed the nature, content, and responsibilities of many careers.	9.4.12.DC.7: Evaluate the influence of digital communities on the nature, content and responsibilities of careers, and other aspects of society 9.4.12.DC.8: Explain how increased network connectivity and computing capabilities of everyday objects allow for innovative technological approaches to climate protection
Digital tools differ in features, capacities, and styles. Knowledge of different digital tools is helpful in selecting the best tool for a given task.	9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task. 9.4.12.TL.2: Generate data using formula-based calculations in a spreadsheet and draw conclusions about the data.
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments 9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem
Interdisciplinary Connections	
MA.K-12.1-8: Integrate mathematical skills using variables, structure, and reasoning to create computer programs.	

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Modifications, Accommodations, and Differentiation			
English Language Learners	IEP / 504	At Risk Students	Gifted and Talented
Scaffolding Word walls Sentence/paragraph frames Bilingual dictionaries/translation Think alouds Read alouds Highlight key vocabulary Annotation guides Think-pair- share Visual aides Modeling In-Class-Support Graphic Organizers Note-taking guides Clarify assignments, directions and instructions Extended time Collaboration between ESL and mainstream classroom teachers	Word walls Visual aides Graphic organizers Multimedia Leveled readers Assistive technology Notes/summaries Extended time Answer masking Answer eliminator Highlighter Color contrast In-Class-Support Provide Study Guides Establish Timelines Clarify Assignments, directions and instructions Parent/teacher communication Assist students with steps to complete assignments Provide student with clear expectations and grading criteria Establish procedures for accommodations /modifications for assessments	Teacher tutoring Peer tutoring Study guides Graphic organizers Extended time Parent communication Modified assignments Counseling Verbal reminders Check student independent work Assist student with planning of assignments Provide student with clear expectations and grading criteria Restate and clarify directions and questions	Challenge assignments Enrichment activities Independent research/inquiry Collaborative teamwork Higher level questioning Critical/Analytical thinking tasks Self-directed activities Open forums and debates in the classroom regarding controversial issues

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 1 - Beginning in Computer Science	Time Frame - 4 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Beginning in Computer Science

This unit introduces students to the basics of Computer Science and the Python programming language.

Core Idea	Performance Expectation (Standard)
Choices individuals make about how and where data is organized and stored affects cost, speed, reliability, accessibility, privacy, and integrity.	8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored
A computing system involves interaction among the user, hardware, application software, and system software.	8.1.12.CS.2: Model interactions between application software, system software, and hardware. 8.1.12.CS.3: Compare the functions of application software, system software, and hardware.
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Explain how technology is used to solve human problems	Lesson 1: What is Computer Science?	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Silly Sentence Project
Describe the process of translating code into commands using a compiler	Lesson 2: Using Python Lesson 3: First Program	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Silly Sentence Project
Explain the relationship between input devices, output devices, CPU, main memory and secondary memory	Lesson 4: Hardware and Software Lesson 5: Output Lesson 6: Input	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Silly Sentence Project
Identify when and how to use variables, numbers, and strings in Python	Lesson 8 - Defining Variables Lesson 9 - Applying Variables Lesson 7: Data Types and Variables Lesson 8: Analog vs. Digital Lesson 9: Understanding Binary	Classroom Assignments and Activities, Programming Labs, Unit 1 Quiz, Unit 1 Test, Silly Sentence Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 2 - Number Calculations and Data	Time Frame - 4 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Number Calculations and Data	
This unit focuses on working with different data types and performing number calculations in Python.	
Core Idea	Performance Expectation (Standard)
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Choices individuals make about how and where data is organized and stored affects cost, speed, reliability, accessibility, privacy, and integrity.	8.1.12.DA.3: Translate between decimal numbers and binary numbers. 8.1.12.DA.4: Explain the relationship between binary numbers and the storage and use of data in a computing device
Digital tools differ in features, capacities, and styles. Knowledge of different digital tools is helpful in selecting the best tool for a given task.	9.4.12.TL.2: Generate data using formula-based calculations in a spreadsheet and draw conclusions about the data.
Career planning requires purposeful planning based on research, self-knowledge, and informed choices.	9.2.12.CAP.6: Identify transferable skills in career choices and design alternative career plans based on those skills.

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Describe the history of the computer over time	Lesson 1: Computer History	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Room Area Project
Define and utilize float, operators and assignments	Lesson 2: Basic Calculations	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Room Area Project
List the symbols for basic operators, exponents, and the order of operations	Lesson 3: Modular Division	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Room Area Project
Define and utilize import, functions, and modules	Lesson 4: Built-in Functions Lesson 5: Random Numbers	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Room Area Project
Apply number calculations and basic functions to real-world data and create charts and graphs	Lesson 6: Big Data Lesson 7: Working with a Real Data Set	Classroom Assignments and Activities, Programming Labs, Unit 2 Quiz, Unit 2 Test, Room Area Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 3 - Making Decisions	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Making Decisions

In this unit, students will explore the different ways to use logic and make decisions in programs.

Core Idea	Performance Expectation (Standard)
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Digital communities influence many aspects of society, especially the workforce. The increased connectivity between people in different cultures and different career fields have changed the nature, content, and responsibilities of many careers.	9.4.12.DC.7: Evaluate the influence of digital communities on the nature, content, and responsibilities of careers, and other aspects of society
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.3: Analyze the effectiveness of the process and quality of collaborative environments.

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Define and code if statements, define-syntax, and list the symbols for relational operators	Lesson 1: Functions Lesson 2: Simple Ifs	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Chatbot Project
Explain and use the Boolean operators (and, not, or)	Lesson 3: Logical Operators	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Chatbot Project
Define and code if else and else if statements in python	Lesson 4: If – Else Lesson 5: Else – If	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Chatbot Project
Create an algorithm the uses both if statements and Boolean conditions	Lesson 6: Defining Algorithms Lesson 7: Algorithm Challenge	Classroom Assignments and Activities, Programming Labs, Unit 3 Quiz, Unit 3 Test, Chatbot Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 4 - Repetition and Loops	Time Frame - 5 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Repetition and Loops

In Unit 4, students will explore different ways of repeating code using loops.

Core Idea	Performance Expectation (Standard)
The usability, dependability, security, and accessibility of devices within integrated systems are important considerations in their design as they evolve.	8.1.12.CS.1: Describe ways in which integrated systems hide underlying implementation details to simplify user experiences.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs.	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables.
Laws govern the use of intellectual property and there are legal consequences to utilizing or sharing another's original works without permission or appropriate credit	9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content.
Digital tools differ in features, capacities, and styles. Knowledge of different digital tools is helpful in selecting the best tool for a given task.	9.4.12.TL.1: Assess digital tools based on features such as accessibility options, capacities, and utility for accomplishing a specified task.

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Define and code while loops and utilize loop control variables	Lesson 1: Loops Lesson 2: Count Variables Lesson 3: Two Ways to End a Loop	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, Divisible by 3 Project
Define and code the count and sum variables in a program	Lesson 4: Data Revisited Lesson 5: Review - Looping Lesson 6: Range Function	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, Divisible by 3 Project
Create for loops with various parameters in the range function	Lesson 7: For Loops Lesson 8: Counting by Other Than 1	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, Divisible by 3 Project
Write a program to add up the prices of items a user types in using the sum function and trace code	Lesson 9: Summing Lesson 10: Review of Algorithms and Tracing Lesson 11: Modeling and Simulation	Classroom Assignments and Activities, Programming Labs, Unit 4 Quiz, Unit 4 Test, Divisible by 3 Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 5 - Programming in EarSketch	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Programming in EarSketch

In this unit, students will explore the complete set of processes and techniques that are involved in using the Python programming language to create music in EarSketch.

Core Idea	Performance Expectation (Standard)
Successful troubleshooting of complex problems involves multiple approaches including research, analysis, reflection, interaction with peers, and drawing on past experiences.	8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Laws govern the use of intellectual property and there are legal consequences to utilizing or sharing another's original works without permission or appropriate credit.	9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content
Laws govern many aspects of computing, such as privacy, data, property, information, and identity.	9.4.12.DC.3: Evaluate the social and economic implications of privacy in the context of safety, law, or ethics

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Read through code to find errors and predict results	Lesson 1: Getting Started with EarSketch Lesson 2: The Building Blocks of a Program Lesson 3: Debugging and Documenting	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Ringtone Project
Create for loops with various parameters in the range function	Lesson 4: Effects in EarSketch Lesson 5: Effects and Envelopes Lesson 6: Tempo and Pitch	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Ringtone Project
Evaluate global and federal laws regarding music sampling and reproduction	Lesson 7: Copyright Lesson 8: Evaluating Correctness Lesson 9: Musical Form and Custom Functions Lesson 10: Recording and Uploading Sounds	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Ringtone Project
Create audio loops in EarSketch (Digital Audio Workstation)	Lesson 11: Making Custom Beats Lesson 12: Looping	Classroom Assignments and Activities, Programming Labs, Unit 5 Quiz, Unit 5 Test, Ringtone Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

	Lesson 13: String Operations	
--	------------------------------	--

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 6 - Graphics	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Graphics	
In this unit students will use x and y coordinates to draw and animate images using the Python programming language.	
Core Idea	Performance Expectation (Standard)
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources and tools.	8.1.12.AP.7: Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users. 8.1.12.AP.9: Collaboratively document and present design decisions in the development of complex programs
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Define and code utilizing the x and y coordinates, frames, and the def function	Lesson 1: Color Code Lesson 2: Colors and Loops Lesson 3: X & Y Coordinates	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Animation Project
Create drawings using canvas.draw_line and draw polygon functions	Lesson 4: Lines Lesson 5: Draw a House	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Animation Project
Create a kaleidoscope in Google's Made w/ Code	Lesson 6: Circles Lesson 7: Emoticons	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Animation Project
Create an animation with 2 circles and 3 lines	Lesson 8: Animations	Classroom Assignments and Activities, Programming Labs, Unit 6 Quiz, Unit 6 Test, Animation Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 7 - Functions	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Functions	
In this unit students will explore how complex processes can be broken down into easily managed steps using functions.	
Core Idea	Performance Expectation (Standard)
Individuals evaluate and select algorithms based on performance, reusability, and ease of implementation.	8.1.12.AP.1: Design algorithms to solve computational problems using a combination of original and existing algorithms.
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed.	9.4.12.CT.1: Identify problem-solving strategies used in the development of an innovative product or practice

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Examine how functions are used efficiently in a program and create their own	Lesson 1: What are Functions? Lesson 2: Creating Functions	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Calendar Project
Define and utilize parameters in a function and return a value	Lesson 3: Parameters Lesson 4: Returning Values	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Calendar Project
Create a program that implements several functions to provide output	Lesson 5: Using Several Functions	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Calendar Project
Examine code and determine where errors may arise on execution	Lesson 6: Tracing Code	Classroom Assignments and Activities, Programming Labs, Unit 7 Quiz, Unit 7 Test, Calendar Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 8 - Lists	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Lists

In this unit, students will analyze how lists give us the ability to store many pieces of information and organize information in a structure that can be easily searched and sorted.

Core Idea	Performance Expectation (Standard)
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Choices individuals make about how and where data is organized and stored affects cost, speed, reliability, accessibility, privacy, and integrity.	8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas
Collaborative digital tools can be used to access, record, and share different viewpoints and to collect and tabulate the views of groups of people.	9.4.12.TL.4: Collaborate in online learning communities or social networks or virtual worlds to analyze and propose a resolution to a real-world problem

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Write a program that declares a list and then adds input from the keyboard	Lesson 1: What are Lists? Lesson 2: Declaring Lists Lesson 3: Element vs Index	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Organizer Project
Use a for loop to automatically store certain data into a list	Lesson 4: For Loops and Lists Lesson 5: List Methods Lesson 6: Lists as Parameters	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Organizer Project
Utilize sort, extend, insert, pop and remove in a program	Lesson 7: Lists and Data	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Organizer Project
Write a program that performs a search and a sort	Lesson 8: Writing a Simple Search Lesson 9: Writing a Simple Sort	Classroom Assignments and Activities, Programming Labs, Unit 8 Quiz, Unit 8 Test, Organizer Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 9 - 2D Lists	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: 2D Lists	
In this unit, students will analyze how 2D (2- dimensional) lists give us the ability to store many pieces of information and access related data points.	
Core Idea	Performance Expectation (Standard)
Programmers choose data structures to manage program complexity based on functionality, storage, and performance trade-offs	8.1.12.AP.2: Create generalized computational solutions using collections instead of repeatedly using simple variables
Choices individuals make about how and where data is organized and stored affects cost, speed, reliability, accessibility, privacy, and integrity.	8.1.12.DA.2: Describe the trade-offs in how and where data is organized and stored.
With a growth mindset, failure is an important part of success.	9.4.12.CI.1: Demonstrate the ability to reflect, analyze, and use creative skills and ideas
There are strategies to improve one’s professional value and marketability.	9.2.12.CAP.3: Investigate how continuing education contributes to one's career and personal growth.

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Describe how information is stored in a two-dimensional list	Lesson 1: What is a 2D List? Lesson 2: Declaring 2D Lists	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Matrices Project
Write a program using loops to process two-dimensional lists	Lesson 3: For Loops and 2D Lists	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Matrices Project
Understand how tracing can be used to predict the outcome of a program	Lesson 4: Algorithms Lesson 5: Tracing Code 2D	Classroom Assignments and Activities, Programming Labs, Unit 9 Quiz, Unit 9 Test, Matrices Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 10 - Programming in EarSketch	Time Frame - 3 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Programming in EarSketch

This unit encompasses the logical elements of programming in EarSketch and evaluating conditions.

Core Idea	Performance Expectation (Standard)
Successful troubleshooting of complex problems involves multiple approaches including research, analysis, reflection, interaction with peers, and drawing on past experiences.	8.1.12.CS.4: Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and fix errors.
Complex programs are designed as systems of interacting modules, each with a specific role, coordinating for a common overall purpose. Modules allow for better management of tasks.	8.1.12.AP.5: Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
Laws govern the use of intellectual property and there are legal consequences to utilizing or sharing another's original works without permission or appropriate credit.	9.4.12.DC.1: Explain the beneficial and harmful effects that intellectual property laws can have on the creation and sharing of content
Collaboration with individuals with diverse experiences can aid in the problem-solving process, particularly for global issues where diverse solutions are needed.	9.4.12.CT.2: Explain the potential benefits of collaborating to enhance critical thinking and problem solving

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Locate errors in code and debug scripts by using the print function and observing output	Lesson 1: Debugging Logic	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Song Project
Evaluate code and assess its completeness and correctness	Lesson 2: Evaluating Correctness	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Song Project
Use algorithms and Booleans in the EarSketch script	Lesson 3: Console Input and Conditions	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Song Project
Explain how list operations are used in EarSketch	Lesson 4: Data Structures Lesson 5: Randomness	Classroom Assignments and Activities, Programming Labs, Unit 10 Quiz, Unit 10 Test, Song Project

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

Unit Name	Unit 11 - Internet	Time Frame - 4 Weeks
Instructional Materials and Resources	Project Stem Platform, CodeSkulptor (online IDE), Replit.com (online IDE), Chromebooks, Video Tutorials/Demonstrations, Classwork Exercises, Software Applications, Internet Research, and Activities.	

Disciplinary Concept: Internet

This unit discusses the rules that govern the internet and keep it running, the types of attacks the internet faces daily such as hacking and Distributed Denial of Service attacks, and how the internet is protected and kept running. The unit will also cover how the internet can be accessed wirelessly through the use of radio waves.

Core Idea	Performance Expectation (Standard)
The scalability and reliability of the Internet are enabled by the hierarchy and redundancy in networks.	8.1.12.NI.1: Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.
Network security depends on a combination of hardware, software, and practices that protect data while it is at rest, in transit, and in use.	8.1.12.NI.2: Evaluate security measures to address various common security threats. 8.1.12.NI.3: Explain how the needs of users and the sensitivity of data determine the level of security implemented.
Laws govern many aspects of computing, such as privacy, data, property, information, and identity. These laws can have beneficial and harmful effects.	9.4.12.DC.4: Explain the privacy concerns related to the collection of data (e.g., cookies) and generation of data through automated processes that may not be evident to users
Digital communities influence many aspects of society, especially the workforce. The increased connectivity between people has changed the nature, content, and responsibilities of many careers.	9.4.12.DC.8: Explain how increased network connectivity and computing capabilities of everyday objects allow for innovative technological approaches to climate protection

Student Learning Objectives (Knowledge and Skills)	Suggested Tasks/Activities	Evidence of Learning (Assessments)
Explain the process and components of transferring data on the internet	Lesson 1: What is the Internet?	Classroom Assignments and Activities, Programming Labs, Unit 11 Quiz, Unit 11 Test, Webpage Project
Define the characteristics and relationship between a DNS and IP address	Lesson 2: IP Addressing and DNS	Classroom Assignments and Activities, Programming Labs, Unit 11 Quiz, Unit 11 Test, Webpage Project
Describe the relationship between packets and routers and their role in communication	Lesson 3: Packets and Routers	Classroom Assignments and Activities, Programming Labs, Unit 11 Quiz, Unit 11 Test, Webpage Project
Code, style, and format elements to manipulate web pages	Lesson 4-6: Making Web Pages	Classroom Assignments and Activities, Programming Labs, Unit 11 Quiz, Unit 11 Test, Webpage Project
Describe data vulnerabilities and cybersecurity	Lesson 7: Cybersecurity	Classroom Assignments and Activities, Programming

Carlstadt-East Rutherford Regional School District
Career and Technology Education Department
Introduction to Computer Science Curriculum

threats	Lesson 8: Net Neutrality	Labs, Unit 11 Quiz, Unit 11 Test, Webpage Project
----------------	--------------------------	---