

Fiero Code Curriculum and NYS CS Standards

| Fiero Course Units | NYS CS Standard Identifier | Standard |
|---|----------------------------|---|
| Javascript Learn sequences, variables, conditionals, loops, functions, and objects as you create a tool to help people do their chores. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| | 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |
| | 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |

| | | |
|---|------------|--|
| | 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| | 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| | 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| | 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| | 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| | 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| Python Learn sequences, variables, conditionals, loops, functions, and objects as you create a birthday card program. | 4-6.CT.1 | Develop a computational model of a system that shows changes in output when there are changes in inputs. |
| | 7-8.CT.1 | Compare the results of alternative models or simulations to determine and evaluate how the input data and assumptions change the results. |
| | 9-12.CT.1 | Create a simple digital model that makes predictions of outcomes. |

| | |
|-----------|---|
| 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |
| 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |
| 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |

| | | |
|---|------------|--|
| | 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| | 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| | 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| Code Blocks Learn sequences, variables, loops, conditionals and events as you create a platformer game. Also practice remixing existing projects. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| | 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |

| | |
|-----------|---|
| 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |
| 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |

| | | |
|--|------------|--|
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| Sphero Learn basic programming concepts like loops, conditionals and functions. Also learn how hardware interacts with software. | 4-6.CT.1 | Develop a computational model of a system that shows changes in output when there are changes in inputs. |
| | 7-8.CT.1 | Compare the results of alternative models or simulations to determine and evaluate how the input data and assumptions change the results. |
| | 9-12.CT.1 | Create a simple digital model that makes predictions of outcomes. |
| | 4-6.CT.2 | Collect digital data related to a real-life question or need. |
| | 7-8.CT.2 | Collect and use digital data in a computational artifact. |
| | 9-12.CT.2 | Collect and evaluate data from multiple sources for use in a computational artifact. |
| | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.6 | Compare two or more algorithms and discuss the advantages and disadvantages of each for a specific task. |
| | 7-8.CT.6 | Design, compare, and refine algorithms for a specific task or within a program. |
| | 9-12.CT.6 | Demonstrate how at least two classic algorithms work, and analyze the trade-offs related to two or more algorithms for completing the same task. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |

| | |
|------------|--|
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |

| | | |
|--|------------|---|
| | 7-8.NSD.2 | Design a project that combines hardware and software components. |
| | 9-12.NSD.2 | Explain the levels of interaction existing between the application software, system software, and hardware of a computing system. |
| | 4-6.NSD.3 | Determine potential solutions to solve hardware and software problems using common troubleshooting strategies. |
| | 7-8.NSD.3 | Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide. |
| | 9-12.NSD.3 | Develop and communicate multistep troubleshooting strategies others can use to identify and fix problems with computing devices and their components. |
| Ozobot Learn basic programming concepts like loops, conditionals and functions. Also learn how hardware interacts with software. | 4-6.CT.1 | Develop a computational model of a system that shows changes in output when there are changes in inputs. |
| | 7-8.CT.1 | Compare the results of alternative models or simulations to determine and evaluate how the input data and assumptions change the results. |
| | 9-12.CT.1 | Create a simple digital model that makes predictions of outcomes. |
| | 4-6.CT.2 | Collect digital data related to a real-life question or need. |
| | 7-8.CT.2 | Collect and use digital data in a computational artifact. |
| | 9-12.CT.2 | Collect and evaluate data from multiple sources for use in a computational artifact. |
| | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |

| | |
|-----------|---|
| 4-6.CT.6 | Compare two or more algorithms and discuss the advantages and disadvantages of each for a specific task. |
| 7-8.CT.6 | Design, compare, and refine algorithms for a specific task or within a program. |
| 9-12.CT.6 | Demonstrate how at least two classic algorithms work, and analyze the trade-offs related to two or more algorithms for completing the same task. |
| 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |

| | |
|------------|--|
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| 7-8.NSD.2 | Design a project that combines hardware and software components. |
| 9-12.NSD.2 | Explain the levels of interaction existing between the application software, system software, and hardware of a computing system. |
| 4-6.NSD.3 | Determine potential solutions to solve hardware and software problems using common troubleshooting strategies. |
| 7-8.NSD.3 | Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide. |
| 9-12.NSD.3 | Develop and communicate multistep troubleshooting strategies others can use to identify and fix problems with computing devices and their components. |

| | | |
|--|------------|--|
| JS Web Learn how to access the contents of a web page with Javascript through the DOM API. Create interactive events and animations as you create a website project. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| Animation Learn basic web animations using Javascript as you create a marketing website. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |

| | | |
|---|------------|--|
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| Code Blocks Games Use variables, loops, functions, conditionals, and events as you create three different games in Scratch. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| | 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |
| | 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |

| | |
|------------|--|
| 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |

| | | |
|---|------------|--|
| Drawing Learn how to use the HTML Canvas to draw on a webpage with Javascript, and animate your drawings as you create an artistic webpage. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| | 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |
| | 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| JS Games Use variables, loops, functions, conditionals, methods, and debugging skills as you build four classic arcade games. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |

| | |
|-----------|---|
| 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |

| | | |
|---|------------|--|
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| SQL Learn about databases, create and edit tables, and run queries as you gain the skills needed to isolate a single piece of data from a large data set. | 4-6.CT.2 | Collect digital data related to a real-life question or need. |
| | 7-8.CT.2 | Collect and use digital data in a computational artifact. |
| | 9-12.CT.2 | Collect and evaluate data from multiple sources for use in a computational artifact. |
| | 4-6.CT.3 | Visualize a simple data set in order to highlight relationships and persuade an audience. |
| | 7-8.CT.3 | Refine and visualize a data set in order to persuade an audience. |
| | 9-12.CT.3 | Refine and visualize complex data sets to tell different stories with the same data set. |
| Makey Makey Learn about computer systems and how hardware interacts with software. Use basic programming skills, like sequences and events, to program the Makey Makey. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| | 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |

| | |
|------------|--|
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| 7-8.NSD.2 | Design a project that combines hardware and software components. |
| 9-12.NSD.2 | Explain the levels of interaction existing between the application software, system software, and hardware of a computing system. |
| 4-6.NSD.3 | Determine potential solutions to solve hardware and software problems using common troubleshooting strategies. |
| 7-8.NSD.3 | Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide. |
| 9-12.NSD.3 | Develop and communicate multistep troubleshooting strategies others can use to identify and fix problems with computing devices and their components. |

| | | |
|--|-----------|---|
| Raspberry Pi Learn about computer systems and how hardware interacts with software through sensors, cameras, and more. | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| | 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| | 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. |
| | 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| | 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| | 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| | 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |

| | | |
|---|------------|--|
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| | 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| | 4-6.NSD.2 | Model how computer hardware and software work together as a system to accomplish tasks. |
| | 7-8.NSD.2 | Design a project that combines hardware and software components. |
| | 9-12.NSD.2 | Explain the levels of interaction existing between the application software, system software, and hardware of a computing system. |
| | 4-6.NSD.3 | Determine potential solutions to solve hardware and software problems using common troubleshooting strategies. |
| | 7-8.NSD.3 | Identify and fix problems with computing devices and their components using a systematic troubleshooting method or guide. |
| | 9-12.NSD.3 | Develop and communicate multistep troubleshooting strategies others can use to identify and fix problems with computing devices and their components. |
| HTML Learn basic HTML syntax and the most popular tags as you create an online job posting. | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |

| | | |
|--|------------|--|
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| | 4-6.DL.4 | Use a variety of digital tools and resources to create and revise digital artifacts. |
| | 7-8.DL.4 | Select and use digital tools to create, revise, and publish digital artifacts. |
| | 9-12.DL.4 | Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts. |
| CSS Learn the basic syntax of CSS and common properties as well as advanced concepts such as positions, grid, and flexbox as you create a library digital display. | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| | 4-6.DL.4 | Use a variety of digital tools and resources to create and revise digital artifacts. |
| | 7-8.DL.4 | Select and use digital tools to create, revise, and publish digital artifacts. |
| | 9-12.DL.4 | Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts. |

| | | |
|---|------------|--|
| Design Learn basic design principles such as a design process, and specific web design principles as you create a website for a fictional town. | 4-6.IC.6 | Identify and explain ways to improve the accessibility and usability of a computing device or software application for the diverse needs and wants of users. |
| | 7-8.IC.6 | Assess the accessibility of a computing device or software application in terms of user needs. |
| | 9-12.IC.6 | Create accessible computational artifacts that meet standard compliance requirements or otherwise meet the needs of users with disabilities. |
| | 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| | 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| | 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| | 7-8.NSD.1 | Design a user interface for a computing technology that considers usability, accessibility, and desirability. |
| Careers Learn about various careers in computer science, and walk through the steps of resume creation, portfolio creation, and interview prep. | 4-6.IC.7 | Identify a diverse range of role models in computer science. |
| | 7-8.IC.7 | Explore a range of computer science-related career paths. |
| | 9-12.IC.7 | Investigate the use of computer science in multiple fields. |

| | | |
|--|--|---|
| <p>Fiero Missions</p> <p>In Fiero, we have over 50 projects for students to work on. These vary by coding discipline (web, games, scripting), and are opportunities for students to use their own creativity to complete the project.</p> | 4-6.CT.4 | Decompose a problem into smaller named tasks, some of which can themselves be decomposed into smaller steps. |
| | 7-8.CT.4 | Write a program using functions or procedures whose names or other documentation convey their purpose within the larger task. |
| | 9-12.CT.4 | Implement a program using a combination of student-defined and third-party functions to organize the computation. |
| | 4-6.CT.5 | Identify and name a task within a problem that gets performed multiple times while solving that problem, but with slightly different concrete details each time. |
| | 7-8.CT.5 | Identify multiple similar concrete computations in a program, then create a function to generalize over them using parameters to accommodate their differences. |
| | 9-12.CT.5 | Modify a function or procedure in a program to perform its computation in a different way over the same inputs, while preserving the result of the overall program. |
| | 4-6.CT.7 | Identify pieces of information that might change as a program or process runs. |
| | 7-8.CT.7 | Design or remix a program that uses a variable to maintain the current value of a key piece of information. |
| | 9-12.CT.7 | Design or remix a program that utilizes a data structure to maintain changes to related pieces of data. |
| | 4-6.CT.8 | Develop algorithms or programs that use repetition and conditionals for creative expression or to solve a problem. |
| 7-8.CT.8 | Develop or remix a program that effectively combines one or more control structures for creative expression or to solve a problem. | |

| | |
|------------|--|
| 9-12.CT.8 | Develop a program that effectively uses control structures in order to create a computer program for practical intent, personal expression, or to address a societal issue. |
| 4-6.CT.9 | Explain each step of an algorithm or program that includes repetition and conditionals for the purposes of debugging. |
| 7-8.CT.9 | Read and interpret code to predict the outcome of various programs that involve conditionals and repetition for the purposes of debugging. |
| 9-12.CT.9 | Systematically test and refine programs using a range of test cases, based on anticipating common errors and user behavior. |
| 4-6.CT.10 | Describe the steps taken and choices made to design and develop a solution using an iterative design process. |
| 7-8.CT.10 | Document the iterative design process of developing a computational artifact that incorporates user feedback and preferences. |
| 9-12.CT.10 | Collaboratively design and develop a program or computational artifact for a specific audience and create documentation outlining implementation features to inform collaborators and users. |
| 4-6.DL.2 | Select appropriate digital tools to communicate and collaborate while learning with others. |
| 7-8.DL.2 | Communicate and collaborate with others using a variety of digital tools to create and revise a collaborative product. |
| 9-12.DL.2 | Communicate and work collaboratively with others using digital tools to support individual learning and contribute to the learning of others. |
| 4-6.DL.4 | Use a variety of digital tools and resources to create and revise digital artifacts. |
| 7-8.DL.4 | Select and use digital tools to create, revise, and publish digital artifacts. |

| | | |
|--|-----------|--|
| | 9-12.DL.4 | Independently select advanced digital tools and resources to create, revise, and publish complex digital artifacts or collection of artifacts. |
|--|-----------|--|