Pascack Valley Regional High School District

Pascack Hills High School, Montvale, New Jersey
Pascack Valley High School, Hillsdale, New Jersey

Course Name: Intro to Computer Science and App Development

Born On: August, 2017
Previous Revision: August, 2022
Current Revision: August, 2023
Board Approval: 8/28/23

COURSE DESCRIPTION: Intro to Computer Science and App Development

Intro to Computer Science and App Development is a full-year, 5 credit course that introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.

All computer science courses in the Pascack Valley Regional High School District are designed to address multiple learning styles and needs, and accommodations and modifications are made for students with disabilities, multilingual students, students at risk of failure, gifted and talented students, and students with 504 plans. *Intro to Computer Science and App Development* builds on students' prior experiences with technology and computers, anticipates higher-level computer science concepts and skills that will be learned in *Honors Computer Science* and beyond, and provides enrichment opportunities to challenge students and engage them in rich, interesting tasks. Students are encouraged to analyze data using tools and models to make valid and reliable claims (9.4.12.IML.3), and various technologies and applications are technology are integrated throughout the curriculum.

The Pascack Valley Regional High School Computer Science Department integrates 21st century life and career skills across its courses, with the dual goal of informing students about careers and fields of study that use computer science (9.3.ST.5, 9.3.ST-ET.5 and 9.3.ST-SM.2), and helping students improve the computational thinking skills they will need to in those careers and fields of study (9.2.12.CAP.2). Computer Science courses address the *New Jersey Student Learning Standards for Career Readiness, Life Literacies and Key Skills*, with a particular emphasis on demonstrating the ability to reflect, analyze and use creative skills and ideas (9.4.12.CI.1), investigating new challenges and opportunities for personal growth, advancement and transition (9.4.12.CI.3), identifying problem-solving strategies used in the development of an innovative product or practice (9.4.12.CT.1), and explaining the potential benefits of collaborating to enhance critical thinking and problem solving (9.4.12.CT.2). Computer Science courses also address the *New Jersey Student Learning Standards for English Language Arts Companion Standards*, with a particular focus on following complex multistep procedures (RST.9-10.3/RST.11-12.3), determining the meaning of symbols, key terms, and other domain-specific words and phrases (RST.9-10.4/RST.11-12.3), and translating quantitative or technical information expressed in words into visual forms and translating information expressed visually or mathematically into words (RST.9-10.7). Similarly, the Computer Science Department seeks to support students by providing them with opportunities to use computational thinking skills in interdisciplinary contexts, in contexts that are meaningful to students, and in contexts that attend to the contributions and perspectives of historically marginalized groups. Specifically, computer science courses will look to incorporate, when appropriate, contributions and experiences of people from the LGBTQ+ community and individuals with disabilities, and references to issues of social and cultural relevance, including climate change.

Essential Questions:

How can a creative development process affect the creation of computational artifacts?

How can computing and the use of computational tools foster creative expression?

How can computing extend traditional forms of human expression and experience?

How are vastly different kinds of data, physical phenomena, and mathematical concepts represented on a computer?

How does abstraction help us in writing programs, creating computational artifacts and solving problems?

How can computational models and simulations help generate new understanding and knowledge?

How can computation be employed to help people process data and information to gain insight and knowledge?

How can computation be employed to facilitate exploration and discovery when working with data?

What considerations and trade-offs arise in the computational manipulation of data?

What opportunities do large data sets provide for solving problems and creating knowledge?

How are programs developed to help people, organizations, or society solve problems?

How are programs used for creative expression, to satisfy personal curiosity, or to create new knowledge?

How do computer programs implement algorithms?

How does abstraction make the development of computer programs possible?

How do people develop and test computer programs?

Which mathematical and logical concepts are fundamental to computer programming?

How does computing enhance human communication, interaction, and cognition?

How does computing enable innovation?

What are some potential beneficial and harmful effects of computing?

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit I – Getting Started: Preview and Setup **Time:** 1-2 weeks **Content Statement:** Unit 1 of the course provides a brief overview of the Mobile CSP curriculum, emphasizing its main theme: learning the principles of computer science while building socially useful mobile apps. The hands-on work focuses on setting up the student's environment, including their programming environment and online portfolios. Students are led through the process of creating a Gmail account, registering on the App Inventor site, and setting up their Google sites portfolio. **Enduring Understandings:** – Computing is a creative activity. – Algorithms are used to develop and express solutions	**Key learning items/concepts:** App Inventor Setup (App Inventor) Portfolio (1-2 weeks) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections** - Engage in programming assignments that have applications in science, mathematics, or business - Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	***STUDENTS WILL BE ABLE TO:*** • Apply a creative development process when creating computational artifacts. • Develop an algorithm for implementation in a program. • Create a computational artifact for creative expression. • Develop an algorithm for implementation in a program. [P2] • Express an algorithm in a language. • Evaluate algorithms analytically and empirically for	**NJSLS Content Standards** 8.1.12.AP.7 8.1.12.AP.8 8.1.12.AP.9 8.2.12.NT.1 8.2.12.NT.2 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning Projects (F)	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support multilingual students in their

to computational problems. – Programming enables problem solving, human expression, and creation of knowledge.		efficiency, correctness, and clarity. • Develop a correct program to solve problems. • Collaborate to develop a program. • Explain how programs implement algorithms.	and make use of structure MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3	- Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	first language and in English, providing materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

			RST.11-12.4 **NJSLS-CLKS** **- 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 2 – Introduction to Mobile Apps & Pair Programming **Time:** 3-6 weeks **Content Statement:** Unit 2 provides an introduction to the App Inventor programming platform and the course's first programming project, the I Have a Dream app, a sound board app. Students are introduced to App Inventor's event-driven programming model. Students first work through a guided tutorial that plays an excerpt of a Martin Luther King speech and are then presented with several exercises that challenge them to extend their understanding by solving problems on their own, working in pairs. This is followed later in the unit by several creative mini projects where students are invited to express their own ideas by developing their own	I Have a Dream Tutorial (App Inventor) (1-2 weeks) I Have a Dream Part 1 (App Inventor) (1-2 weeks) I Have a Dream Projects (App Inventor) (1-2 weeks) Where is North (Compass App using App Inventor) (1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections** - Engage in programming assignments that have applications in science, mathematics, or business	***STUDENTS WILL BE ABLE TO:*** • Create a computational artifact for creative expression. • Create a new computational artifact by combining or modifying existing artifacts. • Describe the variety of abstractions used to represent data. • Identify multiple levels of abstractions that are used when writing programs. • Develop an algorithm for implementation	**NJSLS Content Standards** 8.1.12.DA.3 8.1.12.DA.4 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of structure	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning Projects (F)	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support multilingual students in their first language and in English, providing

computational artifacts. Students are also introduced to several important CS Principles themes and topics. Two lessons focus on hardware and software concepts. The big idea of abstraction is introduced. Students get their first look at binary numbers learning how to count in binary and how to view number systems such as binary, hexadecimal and decimal, as instances of the higher-order abstraction of a positional number system. **Enduring Understandings:** – Computing is a creative activity. – Algorithms are used to develop and express solutions to computational problems. – Programming enables problem solving, human expression, and creation of knowledge. - Computing has global impact.	- Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	in a program. • Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge. • Develop a correct program to solve problems. • Explain how programs implement algorithms. • Evaluate the correctness of a program. • Explain how computing innovations affect communication, interaction, and cognition. • Analyze the beneficial and harmful effects of computing.	MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4	- Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

			NJSLS-CLKS **- 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 3 – Creating Graphics & Images Bit by Bit **Time:** 3-6 weeks **Content Statement:** Unit 3 extends the student's mobile programming toolkit to several new App Inventor components and introduces a number of new programming concepts, including the concept of variables, lists and data abstraction. The main app in this unit, The Paint Pot app, a computational version of finger painting, focuses on App Inventor's drawing and painting features and related topics from the CS Principles framework. The app is presented in four parts each of which is followed by a set of creative project exercises and challenges. This unit also introduces two other apps: Magic 8 Ball app, which provides a first introduction to	Paint Pot 1 (App Inventor) (1/2 - 1 week) Paint Pot 1 Projects (App Inventor) (1/2 - 1 week) Paint Pot 2 (App Inventor) (1/2 - 1 week) Paint Pot 2 Projects (App Inventor) (1/2 - 1 week) Magic 8-Ball (Using App Inventor) (1/2 - 1 week) Map Tour (App Inventor and Google Maps Activity Starter) (1/2 - 1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping	***STUDENTS WILL BE ABLE TO:*** • Create a computational artifact using computing tools and techniques to solve a problem. • Collaborate in the creation of computational artifacts. • Describe the variety of abstractions used to represent data. • Explain how binary sequences are used to represent digital data. • Develop an abstraction when writing a program or creating other computational artifacts. • Use models and simulations to represent	**NJSLS Content Standards** 8.1.12.DA.1 8.1.12.DA.5 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of structure	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support

lists, and Map Tour, which demonstrates how to incorporate external data into a mobile app. Unit 3 also extends the student's understanding of binary number systems and introduces students to the idea of a bit as the fundamental unit of data. Through a number of hands-on and interactive activities students explore how bits are used to represent images, and how redundant parity bits can be used to detect simple data transmission errors. These lessons are complemented nicely by a Blown to Bits reading that focuses on digital. **Enduring Understandings:** - Computing is a creative activity. - Abstraction reduces information and detail to facilitate focus on relevant concepts. - Data and information facilitate the creation of knowledge. - Programming enables problem solving, human expression, and creation of knowledge. - Computing has global impact.	**Interdisciplinary/additional connections** - Engage in programming assignments that have applications in science, mathematics, or business - Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	phenomena. • Collaborate when processing information to gain insight and knowledge. • Extract information from data to discover and explain connections, patterns, or trends. • Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information. • Develop a program for creative expression, to satisfy personal curiosity, or to create new knowledge. • Develop a correct program to solve problems. • Collaborate to develop a program. • Use abstraction to manage complexity in programs. • Evaluate the correctness of a program.	MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4	Projects (F) - Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	multilingual students in their first language and in English, providing materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

		• Employ appropriate mathematical and logical concepts in programming. • Explain how computing innovations affect communication, interaction, and cognition.	**NJSLS-CLKS** **- 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 4 – Exploring Computing: Animation, Simulation, & Modeling **Time:** 3-6 weeks **Content Statement:** Unit 4 focuses on animation, simulation and modeling. The Android Mash app introduces the idea of computer simulation with a computational version of the traditional Whack-a-Mole game. The Coin Flip app, which extends over several lessons, introduces the concept of modeling. The activities in Unit 4 build toward multiple enduring understandings as students learn that models use abstractions, such as a pseudo random number generator (PRNG), to represent real word situations (in this case, the flipping of a coin); students learn how PRNG algorithms are used to model randomness inside a computer, such as with	Android Mash (App Inventor) (1/2 - 1 week) Android Mash Projects (App Inventor) (1/2 - 1 week) Coin Flip (App Inventor) (1/2 - 1 week) Coin Flip Projects (App Inventor) (1/2 - 1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections** - Engage in programming assignments that have applications in science, mathematics, or business	***STUDENTS WILL BE ABLE TO:*** • Create a computational artifact for creative expression. • Create a new computational artifact by combining or modifying existing artifacts. • Use computing tools and techniques for creative expression. • Develop an abstraction when writing a program or creating other computational artifacts. • Use models and simulations to represent	**NJSLS Content Standards** 8.1.12.DA.1 8.1.12.DA.5 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of structure	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning Projects (F)	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support multilingual students in their first language and in English, providing

the Coin Flip app; and students extend the app model to represent different types of coins, including a biased coin and a three-sided coin. This is followed by an experimental lesson where an app that repeatedly "flips" a coin is used to assess the quality of App Inventor's PRNG. Students also learn how one's privacy is impacted by developing technology and computing innovations, and they learn the economic, social and cultural effects of computing innovations, such as real world models of the weather and the solar system. **Enduring Understandings:** – Computing is a creative activity. – Abstraction reduces information and detail to facilitate focus on relevant concepts. – Data and information facilitate the creation of knowledge. – Algorithms are used to develop and express solutions to computational problems. – Programming enables problem solving, human expression, and creation of knowledge. – Computing has global impact.	- Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	phenomena. • Use models and simulations to formulate, refine, and test hypotheses. • Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information. • Develop an algorithm for implementation in a program. • Express an algorithm in a language. • Develop a correct program to solve problems. • Use abstraction to manage complexity in programs. • Employ appropriate mathematical and logical concepts in programming. • Explain how	MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4	- Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

		computing innovations affect communication, interaction, and cognition. • Analyze the beneficial and harmful effects of computing. • Explain the connections between computing and economic, social, and cultural contexts	**NJSLS-CLKS - 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 5 – Algorithms & Procedural Abstraction **Time:** 3-6 weeks **Content Statement:** In Unit 5, algorithms and procedures are examined in more detail. The Logo apps introduce the concept of procedural abstraction and students learn to define and use procedures -- named blocks of code that perform a specific task. By encapsulating the algorithms into named procedures and introducing parameters to help generalize the algorithms, students are led to see the advantages of procedural abstraction. In addition to designing and testing their own algorithms, students are also provided an introduction into the analysis of algorithms. Algorithm efficiency is examined for searching and sorting	Logo 1 (App Inventor) (1/2 - 1 week) Logo 2 (App Inventor) (1/2 - 1 week) The Pong Game (App Inventor) (1/2 - 1 week) Debugging Pong (App Inventor) (1/2 - 1 week) Caesar cipher (App inventor) (1/2 - 1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections**	***STUDENTS WILL BE ABLE TO:*** • Develop an abstraction when writing a program or creating other computational artifacts. • Use multiple levels of abstraction to write programs. • Develop an algorithm for implementation in a program. [• Express an algorithm in a language. • Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time. EXCLUSION STATEMENT Any discussion of nondeterministic polynomial (NP) is	**NJSLS Content Standards** 8.1.12.AP.1 8.1.12.AP.5 8.1.12.AP.6 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **Students with special needs:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **Multilingual students:** Teachers and support staff

algorithms, which are analyzed both experimentally and through mathematical concepts such as functions and graphs. The impact section of this unit focuses on the impact that Web searching algorithms have had on our lives. The activities completed in Unit 5 focus on abstraction, algorithms, and programming concepts. **Enduring Understandings:** – Computing is a creative activity. – Abstraction reduces information and detail to facilitate focus on relevant concepts. – Data and information facilitate the creation of knowledge. – Algorithms are used to develop and express solutions to computational problems. – Programming enables problem solving, human expression, and creation of knowledge. – Computing has global impact.	- Engage in programming assignments that have applications in science, mathematics, or business - Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	beyond the scope of this course and the AP Exam. • Explain the difference between solvable and unsolvable problems in computer science. EXCLUSION STATEMENT Determining whether a given problem is solvable or unsolvable is beyond the scope of this course and the AP Exam. • Explain the existence of undecidable problems in computer science. • Evaluate algorithms analytically and empirically for efficiency, correctness, and clarity. • Develop a correct program to solve problems. • Use abstraction to manage complexity in programs. • Evaluate the correctness of a program. • Explain how computing innovations affect	structure MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4	Projects (F) - Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	will work to support multilingual students in their first language and in English, providing materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

		communication, interaction, and cognition. • Explain how people participate in a problem solving process that scales.	**NJSLS-CLKS - 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 6 – Using and Analyzing Data & Information **Time:** 3-6 weeks **Content Statement:** Unit 6 focuses on various aspects of using and manipulating Data, both within mobile apps and on the Web and Internet. The App Inventor lessons in this unit focus on different types of programming data, including variables and structured data, such as lists and databases. Students build apps that involve persistent data, data that persists from one instance of the app to another, and learn how to share data online by using simple Application Programming Interfaces (APIs), such as the Google Fusion table API. This unit's CS Principles lessons focus on the concept of Big Data and its growing importance and its impact on society. Students are also	Presidents Quiz (App Inventor) (1/2 - 1 week) Presidents Quiz Projects (App Inventor) (1/2 - 1 week) Lists of Lists (App Inventor) (1/2 - 1 week) Data Persistence Projects (App Inventor) (1/2 - 1 week) Fusion Table App (App Inventor and Google Fusion Tables) (1/2 - 1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections**	***STUDENTS WILL BE ABLE TO:*** • Create a computational artifact for creative expression. • Create a new computational artifact by combining or modifying existing artifacts. • Use computing tools and techniques for creative expression. • Develop an abstraction when writing a program or creating other computational artifacts. • Use models and simulations to represent	**NJSLS Content Standards** 8.1.12.AP.2 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of structure MP8. Look for	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S) - online learning courses (F) - Group Learning Projects (F)	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support multilingual students in their first language and in English, providing

introduced to the some of the algorithms for processing massive datasets. **Enduring Understandings:** – Computing is a creative activity. – Abstraction reduces information and detail to facilitate focus on relevant concepts. – Data and information facilitate the creation of knowledge. – Algorithms are used to develop and express solutions to computational problems. – Programming enables problem solving, human expression, and creation of knowledge. – Computing has global impact.	- Engage in programming assignments that have applications in science, mathematics, or business - Consider potential programming applications for climate change - Draw on contexts from diverse groups for programming exercises	phenomena. • Use models and simulations to formulate, refine, and test hypotheses. • Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information. • Develop an algorithm for implementation in a program. • Express an algorithm in a language. • Develop a correct program to solve problems. • Use abstraction to manage complexity in programs. • Employ appropriate mathematical and logical concepts in programming. • Explain how	and express regularity in repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4 **NJSLS-CLKS**	- Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

		computing innovations affect communication, interaction, and cognition. • Analyze the beneficial and harmful effects of computing. • Explain the connections between computing and economic, social, and cultural contexts	**- 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2 **- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		

Intro to Computer Science and App Development *– This full-year, 5 credit course introduces students to the central ideas of computer science, inviting students to develop the computational thinking vital for success across multiple disciplines. The course is unique in its focus on creativity and the application of creative processes when developing computational artifacts. Students will design and implement innovative solutions using an iterative process similar to what artists, writers, computer scientists, and engineers use to bring ideas to life.*					
Content/Topic:	**Key Learning Items/Concepts and Pacing Guide**	**Observable Proficiencies and Skills:**	**NJSLS CS & Design Thinking Standards**	**Formative, Summative, Benchmark, and Alternative Assessments**	**Core Instructional and Supplemental Materials/ Modifications and Accommodations**
Unit 7 – Communication Through the Internet **Time:** 3-6 weeks **Content Statement:** Unit 7 focuses on the Internet, one of the big ideas in computer science. The App Inventor lessons in this unit show different ways to use the internet in apps, including the ability to send text messages over Wifi and finding directions via the Google Maps API. The CS Principles lessons focus on the Internet, how it works, how it enables innovation and collaboration, and security concerns for using it. **Enduring Understandings:** – Computing is a creative activity. – Abstraction reduces information and detail to facilitate focus on relevant concepts.	No Texting While Busy (App Inventor) (1/2 - 1 week) My Directions (App Inventor and Google Maps APIs) (1/2 - 1 week) Broadcast Hub (App Inventor) (1/2 - 1 week) **Content-specific modifications and accommodations** - use multiple representations and technology to support conceptual understanding - provide students with skeletons of code and/or utilize flexible grouping **Interdisciplinary/additional connections** - Engage in programming assignments that have applications in science, mathematics, or business - Consider potential	***STUDENTS WILL BE ABLE TO:*** • Create a computational artifact for creative expression. • Create a new computational artifact by combining or modifying existing artifacts. • Use computing tools and techniques for creative expression. • Develop an abstraction when writing a program or creating other computational artifacts. • Use models and simulations to represent phenomena. • Use models and	**NJSLS Content Standards** 8.1.12.NI.1 8.1.12.NI.2 8.1.12.NI.3 8.1.12.NI.4 **NJSLS SMP** MP1. Make sense of problems and persevere in solving them MP2. Construct viable arguments and critique the reasoning of others MP3. Reason abstractly and quantitatively MP4. Model with mathematics MP5. Attend to precision MP6. Use appropriate tools strategically MP7. Look for and make use of structure MP8. Look for and express regularity in	*Suggestion(s):* Students will be assessed regularly throughout this course, with a focus on both conceptual understanding and procedural fluency. Assessment tools may include the following: - quizzes (F) - tests (S) - performance tasks (F/S) - projects (S) - homework (F) - discussions (F) - journals (F) - Form A, B, or C benchmark (B) - alternative assessments (A) - portfolio (F, S)	Selection of primary sources *Suggestion(s):* Blown to Bits textbook (on grade level); App Inventor (on grade level; remediation; advanced); Mobile CSP website (on grade level) **Modifications and Accommodations:** **<u>Students with special needs</u>:** Teachers and support staff will attend to all modifications and accommodations listed in students' IEPs and 504s. Teachers will incorporate manipulatives, extra time, alternative assessments, scaffolding, spiraling, technology, and flexible grouping to support student learning. **<u>Multilingual students</u>:** Teachers and support staff will work to support multilingual students in their first language and in English, providing

– Data and information facilitate the creation of knowledge. – Algorithms are used to develop and express solutions to computational problems. – Programming enables problem solving, human expression, and creation of knowledge. – Computing has global impact.	programming applications for climate change - Draw on contexts from diverse groups for programming exercises	simulations to formulate, refine, and test hypotheses. • Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information. • Develop an algorithm for implementation in a program. • Express an algorithm in a language. • Develop a correct program to solve problems. • Use abstraction to manage complexity in programs. • Employ appropriate mathematical and logical concepts in programming. • Explain how computing innovations affect communication, interaction, and cognition. • Analyze the beneficial and	repeated reasoning 1. Fostering an inclusive Computing Culture 2. Collaborating around Computing 3. Recognizing and Defining Computational Problems 4. Developing and Using Abstractions 5. Creating Computational Artifacts 6. Testing and Refining Computational Artifacts 7. Communicating about Computing **NJSLS for ELA Companion Standards** RST.9-10.3 RST.9-10.4 RST.9-10.7 RST.11-12.3 RST.11-12.4 **NJSLS-CLKS - 21st Century Life and Careers** 9.4.12.CI.1 9.4.12.CI.3 9.4.12.CT.1 9.4.12.CT.2	- online learning courses (F) - Group Learning Projects (F) - Individual Projects (S) - Oral Presentations (S) - Programming assignments (F) - Mazes (F) - Vocabulary Checks (F)	materials and/or resources to support students' understanding. Students will be given additional time, as appropriate, and translation tools will be utilized as needed. **Students at risk of school failure:** Formative and summative data will be used to monitor student success, and students at risk of failure will receive additional supports and services, which may include parent consultation, extra help, and differentiation strategies, including small group instruction, group work, scaffolding, and spiraling. **Gifted and Talented Students:** Students who excel in their mastery of course standards will be further challenged with more complex tasks, extensions of concepts and skills, and extended problem solving and critical thinking opportunities.

		harmful effects of computing. • Explain the connections between computing and economic, social, and cultural contexts	**- Technology** 9.4.12.IML.3 **- Career Education** 9.2.12.CAP.2 9.3.ST.5 9.3.ST-ET.5 9.3.ST-SM.2		