Pascack Valley Regional High School District

Pascack Hills High School, Montvale, New Jersey 07675
Pascack Valley High School, Hillsdale, New Jersey 07642

Course Name: Honors Computer Science

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

Course Description: Honors Computer Science

Honors Computer Science is a full-year programming course using the *Java* language. *Java* is an object-oriented programming language developed by Sun Microsystems. The course also teaches students how to use the *JAVA*™ language's object -oriented technologies to solve problems. Students will learn how to create classes, objects, and applications using the language.

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. *Honors Computer Science* builds on concepts learned and skills developed in *Intro to Computer Science and App Development*, while also spiraling in those concepts and skills to reinforce and strengthen students' knowledge of computer science. Additionally, *Honors Computer Science* anticipates higher-level computer science concepts and skills that will be learned in *AP Computer Science A* and beyond, and enrichment opportunities are provided 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 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.

Honors Computer Science: Honors Computer Science is a full-year programming course using the Java language. Students will learn how to use the Java language's object - oriented technologies to solve problems, create classes, objects, and applications.

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 1 – Introduction (8 – 9 weeks)	**0. Introduction to Computers and Programming Languages** (1/2 week) • A History of Computers • Computer Architecture • Programming Languages • Java • Exercises **1. Introduction to Object-Oriented Programming and Software Development** (2 weeks) • Classes and objects • Messages and Methods • Class and Instance Data Values • Inheritance • Software Engineering and Software Life Cycle • Having Fun with Java • Exercises **2. Java Programming Basics** (2 weeks) • The first Java Application • Program Components • Edit-Compile-Run	- Briefly state a history of computers. - Name and describe four major components of the computer. - Convert binary numbers to decimal numbers and vice versa. - State the difference between the low-level and high-level programming languages. - Name the basic components of object-oriented programming. - Differentiate classes and objects. - Differentiate class and instance methods. - Differentiate class and instance data values. - Draw object diagrams using icons for classes, objects, and other components of object-oriented programming using inspiration software. - Name and explain the stages of the software life cycle. - Identify the basic components of Java programs. - Distinguish two types of Java programs - applications and applets. - Write simple Java applications and applets. - Describe the difference between object declaration and	**NJSLS Content Standards** 8.1.12.DA.2 8.1.12.AP.1 8.1.12.AP.2 8.1.12.AP.3 8.1.12.AP.4 8.1.12.AP.5 8.1.12.AP.6 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 and make use of structure MP8. Look for and express regularity in repeated reasoning	*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) - Individual Projects (S) - Oral Presentations	Selection of primary sources *Suggestion(s):* Horstmann, Cay. *Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level); Trees, Frances P. *AP Computer Science Study Guide to Accompany Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level; remediation; advanced) **Software:** Eclipse SDK, Version: 3.2.2, Apache Software Foundation **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 will work to support multilingual students in

	Cycle **3. Numerical Data** (4 weeks) • Variables • Arithmetic Expressions • Constants • The Math Class **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	object creation. - Describe the process of creating and running Java programs. - Use the Graphics class from the standard Java package. - Select proper types for numerical data. - Write arithmetic expressions in Java. - Evaluate arithmetic expressions using the precedence rules. - Describe how the memory allocation works for objects and primitive data values. - Write mathematical expressions using methods in the Math class.	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 **- 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	(S) - Programming assignments (F)	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.

Honors Computer Science: Honors Computer Science is a full-year programming course using the Java language. Students will learn how to use the Java language's object - oriented technologies to solve problems, create classes, objects, and applications.

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 – **Classes, Selections, and Repetitions** (16-17 weeks)	**4. Defining Instantiable Classes** (4 weeks) • Defining Instantiable Classes • Instantiable Classes and Constructors • Visibility Modifiers: public and private • Local Variables, Return Values, and Parameter passing • Loan Calculator Program with Instantiable Class • Exercises **5. Processing Input with Applets** (4 weeks) • Placing GUI objects on Applets • Adding ActionListener to an Event Source • Absolute Positioning of GUI Objects • The Button Class • Converting text to a numerical Value • Sample Program: Finding Body Mass Index (BMI) **6. Selection Statements** (4	- Apply the incremental development technique in writing programs. - Define an instantiable class with multiple methods and a constructor. - Differentiate the local and instance variables. - Define and use value-returning methods. - Distinguish private and public methods. - Distinguish private and public data members. - Describe how the arguments are passed to the parameters in method definitions. - Use System.out for temporary output to verify the program code. - Define an applet with multiple methods. - Incorporate a simple event-handling routine to an applet to process input. - Use the reserved word this in your programs. - Run applets without using an applet viewer or browser. - Implement selection control in a program using if statements. - Implement selection control in a program using switch	**NJSLS Content Standards** 8.1.12.DA.2 8.1.12.AP.1 8.1.12.AP.2 8.1.12.AP.3 8.1.12.AP.4 8.1.12.AP.5 8.1.12.AP.6 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 and make use of structure MP8. Look for and express regularity in repeated reasoning	*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) - Individual Projects (S) - Oral Presentations (S)	Selection of primary sources *Suggestion(s):* Horstmann, Cay. *Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level); Trees, Frances P. *AP Computer Science Study Guide to Accompany Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level; remediation; advanced) **Software:** Eclipse SDK, Version: 3.2.2, Apache Software Foundation **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.

	weeks) • The if statement • Boolean Expressions and variables • Nested-if statements • ListBox • 6.5 The switch statement • 6.6 Sample Program: Drawing Shapes 6.7 Exercises **7. Repetition Statements** (4 weeks) 7.1 The while statement 7.2 Pitfalls in Writing Repetition Statements 7.3 The do-while Statement 7.4 ResponseBox 7.5 The for statement 7.6 Nested-for statement 7.7 The Format class 7.8 Loan Tables 7.9 Sample Program: Hi-Lo Game 7.10 Recursive methods (optional) 7.11Exercises **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	statements. - Write boolean expressions using relational and boolean operators. - Evaluate given boolean expressions correctly. - Nest an if statement inside another if statement's then or else part correctly. - Choose the appropriate selection control statement for a given task. - Implement repetition control in a program using while statements. - Implement repetition control in a program using do-while statements. - Implement repetition control in a program using for statements. - Nest a loop repetition statement inside another repetition statement. - Choose the appropriate repetition control statement for a given task.	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 **- 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	- Programming assignments (F)	**Multilingual students:** Teachers and support staff 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.

	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				

Honors Computer Science: Honors Computer Science is a full-year programming course using the Java language. Students will learn how to use the Java language's object - oriented technologies to solve problems, create classes, objects, and applications.

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 – **Characters, Arrays, and Sorting** (10-11 weeks)	**8. Characters and Strings** (4 weeks) 8.1 Characters 8.2 Strings 8.3 Primitive verses Reference Types 8.4 StringBuffer 8.5 Passing Objects as Parameters 8.6 Returning an Object from Methods 8.7 Sample Program: Word Play 8.8 Exercises 9. **Arrays** (4 weeks) 9.1 Array Basics 9.2 Array of Objects 9.3 Passing Arrays to Methods 9.4 MultInputBox 9.5 Self-Referencing Pointer 9.6 Sample Development: The Address Book 9.7 Two-Dimensional Arrays 9.8 Vectors 9.9 Exercises **10. Sorting and Searching** (2 weeks) • Searching	- Declare and manipulate data of the char data type. - Write string processing programs using String and StringBuffer objects. - Differentiate the String and StringBuffer classes and use the correct class in solving a given task. - Distinguish the primitive and reference data types and show how the memory allocation between the two is different. - Tell the difference between equality and equivalence testings for String objects. - Show, by using the state-of-memory diagrams, how objects are passed to methods and returned from methods. - Manipulate a collection of data values using an array. - Declare and use an array of primitive data types in writing a program. - Declare and use an array of objects in writing a program. - Describe how a two-dimensional array is implemented as an array of arrays. - Manipulate a collection of	**NJSLS Content Standards** 8.1.12.DA.2 8.1.12.AP.1 8.1.12.AP.2 8.1.12.AP.3 8.1.12.AP.4 8.1.12.AP.5 8.1.12.AP.6 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 and make use of structure MP8. Look for and express regularity in repeated reasoning 1. Fostering an inclusive	*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) - Individual	Selection of primary sources *Suggestion(s):* Horstmann, Cay. *Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level); Trees, Frances P. *AP Computer Science Study Guide to Accompany Java Concepts*. 4th ed. Hoboken, N.J.: Wiley, 2006. (on grade level; remediation; advanced) **Software:** Eclipse SDK, Version: 3.2.2, Apache Software Foundation **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 will work to support multilingual students in

	• Sorting • Sample Program: Sorting an AddressBook • Exercises **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	objects using a vector. - Define a method that accepts an array as its parameter and a method that returns an array. - Perform linear and binary search algorithms on small arrays. - Determine whether a linear or binary search is more effective for a given situation. - Perform selection and bubble sort algorithms. - Describe the heapsort algorithm and show how its performance is superior to the other two algorithms. - Apply basic sorting algorithms to sort an array of objects.	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 **- 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	Projects (S) - Oral Presentations (S) - Programming assignments (F)	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.