Hoboken Public Schools

PLTW Introduction to Computer Science Curriculum

Introduction to Computer Science Curriculum

HOBOKEN PUBLIC SCHOOLS

Course Description

Introduction to Computer Science Design (ICS) is a high school level course for 9th or 10th grade students who are interested in design and engineering. With emphasis on computational thinking and collaboration, this year-long course provides an excellent entry point for students to begin or continue the PLTW Computer Science K-12 experience. Introduction to Computer Science will expose students to a diverse set of computational thinking concepts, fundamentals, and tools, allowing them to gain understanding and build confidence.

In Introduction to Computer Science, students will use visual, block-based programming and seamlessly transition to text-based programming with languages such as Python to create apps and develop websites, and learn how to make computers work together to put their design into practice. They'll apply computational thinking practices, build their vocabulary, and collaborate just as computing professionals do to create products that address topics and problems important to them.

Introduction to Computer Science helps students create a strong foundation to advance to Computer Science Principles, Computer Science A, and beyond.

Course Resources

1:1 computers with MIT App Inventor

Python

Engineering Notebook

Pacing Guide

Unit Titles	Time Frame
Unit One: Mobile Computing	**5 Months**
Unit Two: Crowds and Clouds	**5 Months**

Unit 1 – Mobile Computing

5 Months

The goal of Unit 1 is to excite students about programming and build students' ability to break apart a problem and persistently build solutions in small steps. Student creativity, collaboration, and an iterative design process are emphasized. Students work with MIT App Inventor to create basic apps that rely on the concepts of event-driven programming, branching and iteration, variables, and abstraction – the building blocks of creating with code.

Essential Questions

- How has computing affected the world we live in?
- Why is it advantageous to break a problem down into smaller pieces and build a solution incrementally?
- How do computers represent the data in words, numbers, pictures, and sound?
- How is a complex piece of software organized?
- How do teams plan and create complex solutions to a problem?
- How do I safely use the Internet?
- How do people collaborate to create software applications?

Essential Learning Outcomes

- Students will pick a grand challenge and consider how mobile computing, the Internet, Big Data, and simulation are contributing to solving that challenge.
- Students will use MIT App Inventor (AI2) to create an app with a drawing canvas and its own camera control. The app allows users to draw on photos by dragging and tapping on the screen.
- Students will create a mobile app with a counter operated by buttons and voice recognition. Students learn about the properties and events associated with AI2 components and are introduced to Agile development.
- Students will analyze digital and analog sound. Students use Audacity® software and a spectrum analyzer to create and analyze a digital recording of themselves.
- Students will use an AI2 canvas to create a bouncing ball with sounds that depend on which side the ball bounces against.
- Students will use GIMP to create a sprite from an image. Representation and ownership of images are considered.
- Students will create a game, Sprite Smash, in which a sprite pops up at random positions on the screen. The player scores points by tapping the sprite before it jumps to a new location. Students apply event handlers, procedures, global variables, and the Cartesian coordinate system.
- Students will create an app in which a sprite slides around a canvas based on randomness, tablet tilt, flings, or taps.
- Students will pick a task to complete. A crowdsourced document shared among teachers accumulates tasks in bite sized pieces appropriate for students new to programming. Students may select from that list or branch out into new ground.
- Students will pick a larger goal to complete, written as one or more user stories. Students break the user story into smaller tasks and complete a sprint toward their goal. A crowdsourced document shared among teachers accumulates successful sprints and their decomposition into tasks. Students may select from that list or branch out into new ground.
- Students will interview a family member, a community member, and a school member while seeking a client for a mobile app. Students consider examples of how mobile and embedded computing are improving people's lives, and with what accompanying detriment. We're all engineers. What will you make?
- Students will consider life as one big collaboration. Students reason about consequences for themselves and others in scenarios involving texting, creating and sharing pictures, posting to social media, and using email.
- Students will collaborate to create a product that includes text. The data will include both a text-encoded constrained-response data field and a prose data field. Examples could include a directory of local businesses or organizations, a curated list of websites about student interests, a biodiversity survey of plants and animals, or a compilation of student-written articles, comics, opinion pieces, and advertisements.

- Students will research and present about career opportunities in a field of their choice, focusing on the way in which CS and IT skills improve the opportunities in that career field.
- Students will develop an app to express creativity or to meet a need in a project growing out of the interviews in the previous lesson.

Technology Infusion

- 8.1.12.A.1: Create a personal digital portfolio which reflects personal and academic interests, achievements, and career aspirations by using a variety of digital tools and resource

Standards Addressed

- 9-10.RST.4 - Reading Science/Technical Determine the meaning of symbols, key terms, and other domain-specific words and phrases as they are used in a specific scientific or technical context relevant to grades 9–10 texts and topics
- 9-10.W.1.d – Writing Establish and maintain a formal style and objective tone while attending to the norms and conventions of the discipline in which they are writing.
- 9-10.W.2.e – Writing Establish and maintain a formal style and objective tone while attending to the norms and conventions of the discipline in which they are writing.
- 9-10.SL.1.c - Speaking and Listening Propel conversations by posing and responding to questions that relate the current discussion to broader themes or larger ideas; actively incorporate others into the discussion; and clarify, verify, or challenge ideas and conclusions.
- 9-10.SL.6 - Speaking and Listening Adapt speech to a variety of contexts and tasks, demonstrating command of formal English when indicated or appropriate.
- 9-10.L.1 – Language Demonstrate command of the conventions of standard English grammar and
- usage when writing or speaking. 9-10.L.2 – Language Demonstrate command of the conventions of standard English capitalization, punctuation, and spelling when writing.
- 9-10.L.2.c – Language Spell correctly.
- 9-10.L.4 – Language Determine or clarify the meaning of unknown and multiple-meaning words and phrases based on grades 9–10 reading and content, choosing flexibly from a
- range of strategies.
- 9-10.L.4.a – Language Use context (e.g., the overall meaning of a sentence, paragraph, or text; a word's position or function in a sentence) as a clue to the meaning of a word or phrase.

Differentiation

- **Time**: Extra time for assigned tasks, adjust length of assignment, timeline with due dates for reports and projects, communication system between home and school and provide lecture notes/outline.
- **Processing**: Extra Response time, verbalize steps, repeat, clarify or reword directions, Mini-breaks between tasks, Provide a warning for transitions, and partnering.
- **Recall**: Teacher-made checklist, Use visual graphic organizers, reference resources to promote independence and visual/verbal reminders
- **Tests/Quizzes/Grading:** Extended time, Study guides, shortened tests, and read directions aloud.
- **Behavior/Attention:** Consistent daily structured routine, simple and clear classroom rules, and frequent feedback.
- **Organization:** Individual daily planner, display a written agenda, note-taking assistance, and Color code materials

Assessments

- Engineering notebook
- Portfolio of digital design project
- App Challenge results
- Test/quizzes/vocabulary

21st Century Learning Connection

- 9.3.ST.1: Apply engineering skills in a project that requires project management, process control and quality assurance.
- 9.3.ST-ET.1: Use STEM concepts and processes to solve problems involving design and/or production.
- 9.3.ST-ET.4: Apply the elements of the design process

Unit 2 Crowds and Clouds

5 Months

Unit 2 Overview

The goal of this Unit is for students to become comfortable implementing algorithms using conditionals and loops in Python® and to generalize algorithmic structures from corresponding MIT App Inventor and Python® code. Students create a game simulation, learning about functions, arguments, and return values. Students generalize from this simulation to learn about model abstraction and the impact that simulation and data are having across all career fields. Students then apply their Python® skills to compete in a rock-paper-scissors game, developing functions to implement a complex strategy that attempts to detect their opponent's strategy.

Essential Questions

- How do apps share data across devices through the Internet to let users to interact?
- What data are you contributing via your interactions on the Web and through apps, and to whom are you contributing the data?
- What new phenomena are being created when many users are contributing to a data set?
- How are algorithms used to solve common problems?
- How are functions and abstraction used to handle complexity?
- How are data and simulation affecting career fields?

Essential Learning Outcomes

- Students will explore basic HTML and CSS, the languages of the Web. Students will manipulate a locally stored Web page, adding elements and modifying the background color, reinforcing hexadecimal RGB color representation.
- Students learn how to use an application programming interface (API) to send commands to a Web server over the Web. By using an interface other than a browser, they learn about GET and POST requests over the Web's HTTP protocol.
- Students will use MIT App Inventor to create a simple app to allow a user to send and receive API data over the Web. They automate the sending of data in a cybersecurity challenge.
- Student will use a Google sheet to share data about themselves with the class. Patterns are observed and compared between two groups.
- Students will discuss personally identifiable information (PII) and safe/common/legal practices regarding PII.

- Students will develop an app that shares data across multiple users.
- Students will collect data about outcomes in Ezee, a game in which outcomes are random and players try to get 14 of a kind.
- Students will explore a Python® development environment and become familiar with a code editor and an interactive command line.
- Students will define and call functions with arguments to accomplish simple mathematical tasks.
- Students will compare the meaning of the terms "variable," "function," and "equal" in the contexts of mathematics and computer programming languages.
- Students will learn three patterns for loops: accumulation, aggregations, and finding the maximum or minimum in a set. For each pattern, students study an example, complete an example, and then create their own code.
- Students will create a sequence of Python® functions to simulate a single game of Ezee, which they played at the beginning of the lesson.
- Students will explore a distribution resulting from a Monte Carlo simulation and identify which details of a phenomenon are parameterized and which details are abstracted away by a model.
- Students will research the impact of modeling and simulation in a career field of their choice.
- Students will create an algorithm to analyze a competitor's history in rock-paper-scissors and predict the competitor's next move. Students implement their algorithm in Python® and compete in a round-robin tournament.

Technology Infusion

- 8.2.8C.4: Identify steps in the design process that would be used to solve a designated problem.
- 8.1.12.A.1: Create a digital portfolio which reflects personal and academic interests achievements, and career aspirations by using a variety of digital tools and resources.

Standards Addressed

- AS.W.4 - Writing Produce clear and coherent writing in which the development, organization, and style are appropriate to task, purpose, and audience.
- AS.W.6 - Writing Use technology, including the Internet, to produce and publish writing and to interact and collaborate with others.
- AS.W.10 - Writing Write routinely over extended time frames (time for research, reflection, and revision) and shorter time frames (a single sitting or a day or two) for a range of tasks, purposes, and audiences.
- AS.SL.1 - Speaking and Listening Prepare for and participate effectively in a range of conversations and collaborations with diverse partners, building on others' ideas and expressing their own clearly and persuasively.
- AS.SL.4 - Speaking and Listening Present information, findings, and supporting evidence such that listeners can follow the line of reasoning and the organization, development, and style are appropriate to task, purpose, and audience.
- AS.SL.6 - Speaking and Listening Adapt speech to a variety of contexts and communicative tasks, demonstrating command of formal English when indicated or appropriate.
- AS.L.1 - Language Demonstrate command of the conventions of standard English grammar and usage when writing or speaking.
- AS.L.2 - Language Demonstrate command of the conventions of standard English capitalization, punctuation, and spelling when writing.
- AS.L.6 - Language Acquire and use accurately a range of general academic and domain-specific words and phrases sufficient for reading, writing, speaking, and listening at the college and career readiness level; demonstrate independence in gathering vocabulary knowledge when considering a word or phrase important to comprehension or expression.

Differentiation

- **Time**: Extra time for assigned tasks, adjust length of assignment, timeline with due dates for reports and projects, communication system between home and school and provide lecture notes/outline.
- **Processing**: Extra Response time, verbalize steps, repeat, clarify or reword directions, Mini-breaks between tasks, Provide a warning for transitions, and partnering.
- **Recall**: Teacher-made checklist, Use visual graphic organizers, reference resources to promote independence and visual/verbal reminders
- **Tests/Quizzes/Grading:** Extended time, Study guides, shortened tests, and read directions aloud.
- **Behavior/Attention:** Consistent daily structured routine, simple and clear classroom rules, and frequent feedback.
- **Organization:** Individual daily planner, display a written agenda, note-taking assistance, and Color code materials

Assessments

- Engineering notebook
- Portfolio of digital design project
- App Challenge results
- Test/quizzes/vocabulary
- Test/quizzes/vocabulary

21st Century Learning Connection

- 9.3.ST.1: Apply engineering skills in a project that requires project management, process control and quality assurance.
- 9.3.ST-ET.1: Use STEM concepts and processes to solve problems involving design and/or production.
- 9.3.ST-ET.4: Apply the elements of the design process