PLTW Computer Science Essentials

The following learning targets represent the major concepts studied and assessed in this course. Updated 2022 - 2023

Semester 1

Unit 1.1 Introduction to Computer Science Essentials:

- Mobile computing has changed our world, and most of today's students have never known a life without apps. This lesson gives students the tools they need to create their own apps on Android and iOS using MIT App Inventor. The goal of this lesson is to introduce students to coding fundamentals through block-based programming. Students will develop independent and collaborative strategies that will help them communicate around computing as they learn and reinforce the fundamental concepts of coding. With a powerful yet approachable tool, students will use their creativity to produce computational artifacts like those that are essential to all of us today.

Unit 1.2 Collaborating Around Computing:

- This lesson focuses on collaborative strategies that coding professionals use when creating programs and applications, while it continues to introduce essential concepts in computer science and coding. The lesson also introduces the idea that computer science can be more than just innovation and creative expression; it can be powerful in solving many problems in today's world. Students apply an Agile development process and task decomposition to solve a problem that meets the needs of others.

Unit 1.3 Innovation and Problem-Solving:

- The final lesson of this unit gives students the freedom to select the focus of their development in choosing the type of app they would like to collaborate to create. Student groups will apply development strategies and user-centered research to create an app that has value to others. Students will gain insight into the importance of creativity, persistence, and value of diverse perspectives in an iterative development process.

Unit 2.1 Transitions to Text-Based Coding:

- Block-based programming is a great way to introduce coding fundamentals, but sometimes a highly abstracted programming language can limit the ability to create code the way you want. Lesson 2.1 allows students to continue in a block-based environment and peek into a text-based language.

Unit 2.2 Computing and Careers in our Society:

- Just as clicks of a button or “swipes” on a screen are used to trigger events in an app, today, images are increasingly important as a way to make decisions in programming. In this lesson, students explore image processing and other innovations that are changing our society. They begin to investigate the wide range of careers in computer science and how computational thinking is an important part of the majority of professions today and in the future.

Unit 2.3 Computing in our World:

- Tomorrow’s solutions involve all of us. In the final lesson, students learn how to take collaborations to scale to achieve a common goal. The final lesson challenges students to navigate an environment as part of a team, using images to identify each other and the environment, and make a decision on the best path forward.

Unit 3.1 Text-Based Coding:

- Lesson 3.1 introduces the power of text-based programming using Python®. Students explore and apply the same essential coding fundamentals introduced earlier in block-based coding while learning more about the flexibility Python provides. Students create a point-of-sale program for a restaurant that requires the use of variables, conditionals, and user input to allow a user to order from a menu.

Unit 3.2 Text-Based Solutions:

- In this lesson, students will learn about loops, lists, and data types in Python®, applying those skills to create code to be used in a social media program, then a course registration program, and eventually, a rock-paper-scissors tournament.

Unit 3.3 The Power of Text-Based Programming:

- **In the final unit of this lesson, students collaboratively design, create, and test a program that will solve a problem from one of their other classes by automating a repetitive process of their choosing.**
    - **Examples of the kinds of problem that students might choose include:**
    - **Punnet squares or dichotomous keys from Biology**
    - **Formulas from Physics, Algebra, or Geometry**
    - **Sentence diagramming from English (this type of program can get quite involved)**

Unit 4.1 Innovation of Computational Probelm-Solving:

- **The goal of this lesson is to allow students the opportunity to apply the collaboration, technical, and communication skills that they have developed to solve an authentic problem that is relevant to them.**
- **Part E is deigned for getting authentic feedback from users and complete one last iteration before the final project is completed and presented. Determine the type of presentations you will have your students do to determine how many days will be needed for Part F.**