Park Hill School District

Building Successful Futures • Each Student • Every Day

High School PLTW AP Computer Science Essentials

Course Description: Computer Science Essentials exposes students to a diverse set of computational thinking concepts, fundamentals, and tools, allowing them to gain understanding and build confidence. Students 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 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.

Scope and Sequence:

Time Frame	Unit	Instructional Topics
5 weeks	Creative Computing, Building with Blocks	Topic 1: Creative Computing, Building with Blocks
4 weeks	Computing and Society: Transitions to Text	Topic 1: Computing and Society: Transitions to Text
5 weeks	Solving with Syntax	Topic 1: Using Backend Services
4 weeks	Computing with a Purpose	Topic 1: Innovation of Computational Problem Solving

Curriculum Revision Tracking

Summer 2020

- Although many components of the previous curriculum are included, this was an overall complete revision of the course.

Essential Learning Outcomes:

- ELO 1: Students learn why it is important to become a creator of apps and not just a user.
- ELO 2: Students learn why independent and cooperative strategies are important in computer science.
- ELO 3: Students articulate what makes a computer science concept fundamental or essential. Students understand why decisions in programs are narrowed down to two options such as “yes or no” or “true or false”
- ELO 4: Students demonstrate how the variable scope influences the structure of an algorithm?
- ELO 5: Students reflect on why user stories and user-centered design are so important when creating an app.1.1.4-3
- ELO 6: Students identify what mathematical and logical concepts are used repeatedly in programming, articulate the advantages and benefits of using loops in an algorithm and identify the challenges and advantages of pair programming.
- ELO 7: Students conceptualize the different ways iteration plays a role in a program and in an app that is created for others.
- ELO 8: Students synthesize information related to the topics of: 1) What is the purpose of a program? 2) Where does a program integrate mathematical and/or logical concepts? 3) What does an algorithm do in the program? and 4) What problems are really worth the effort to try to solve?
- ELO 9: Students learn what are some of the ways concepts in blocks are represented in languages like Python® and JavaScript™ and how realistic it is to expect coding professionals to be experts on all programming languages.
- ELO 10: Students identify the challenges to programming a vehicle to navigate an environment even if it is well defined.
- ELO 11: Students demonstrate their understanding of why different languages are sometimes better suited for expressing different algorithms and how abstractions manage complexity in a program.
- ELO 12: Students explain the advantages and disadvantages of working at high and low levels of abstraction, what coding constructs in blocks look like in text-based language, and (Loops) why arrays are an essential concept in programs.
- ELO 13: Students demonstrate their understanding of the role of the Scrum on a development team, what the purpose of your program is, where your program integrates with mathematical and/or logical concepts and what does one of the algorithms in your program do.
- ELO 14: Students learn how mobile, wireless, and networked computing had an impact on innovation throughout the world.

- ELO 15: Students understand some advantages and challenges of cloud computing, how abstraction in the programming language being used is managing complexity in my program and how to apply independent, cooperative, and collaborative strategies to find programming answers.
- ELO 16: Students can describe what an algorithm does to someone new to coding and identify some mathematical and logical concepts that are used over and over.

Unit 1: Creative Computing, Building with Blocks

Subject: PLTW Computer Science Essentials
Grade: 9-12
Name of Unit: Creative Computing, Building with Blocks
Length of Unit: 5 weeks
Overview of Unit: Mobile computing has changed our world, and most of today's students have never known a life without apps. This unit gives students the tools they need to create their own apps using MIT App Inventor. The goal of this unit 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.

Topic 1: Creative Computing, Building with Blocks

Activity	Title	ELO
Activity 1.1.1	Getting Started with Block-based Programming: Digital Doodle	1, 2
Activity 1.1.2	Algorithms and Coding Fundamentals: Happy Accelerometer	3
Activity 1.1.3	Conditionals and Event-driven Programming: Happy Balance	4,5
Activity 1.1.4	Local and Global Variables: Guessing Game-Two Player	5,6
Activity 1.1.5	Iteration and Loops: Guessing Game-One Player	6, 7
Activity 1.1.6	App Development: Creative Expression	7, 8

Unit 2: Computing and Society: Transitions to Text

Subject: PLTW Computer Science Essentials
Grade: 9-12
Name of Unit: Computing and Society: Transitions to Text
Length of Unit: 4 weeks
Overview of unit: Within this unit Students will learn how to use block-based programming to introduce coding fundamentals. Students will learn a highly abstracted programming language can sometimes limit the ability to create code in some constructs, thus students learn to use a block-based environment to peek into a text-based language.

Topic 1: App Navigation

Activity	Title	ELO
Activity 2.1.1	Transitioning from Blocks to Text	9, 10
Activity 2.1.2	Coding Fundamentals: Dead Reckoning	11
Activity 2.1.3	Lists	12
Activity 2.1.4	Two-dimensional Lists	13
Activity 2.1.5	Map It, Drive It.	14

Unit 3: Solving with Syntax

Subject: PLTW Computer Science Essentials
Grade: 9-12
Name of Unit: Solving with Syntax
Length of Unit: 5 weeks
Overview of Unit: Within Unit 3, students are introduced to the power of programming in a cloud computing environment using the text-based language *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.

Topic 1: Using Backend Services

Activity	Title	ELO
Activity 3.1.1	Python Programming	15
Activity 3.1.2	Variables and Conditionals	16
Activity 3.1.3	Combo Menu	16

Unit 4: Computing with a Purpose

Subject: PLTW Computer Science Essentials
Grade: 9-12
Name of Unit: Computing with a Purpose
Length of Unit: 4 weeks
Overview of Unit: The goal of this unit 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. Students will get authentic feedback from users and complete one last iteration before presenting the final project presentation.

Topic 1: Innovation of Computational Problem Solving

Activity	Title	ELO
Activity 4.1.1	Introduction and Overview	13
Activity 4.1.2	Find an Idea Worth Pursuing	13
Activity 4.1.3	Capture Development Milestones	13
Activity 4.1.4	Prepare, Investigate, and Plan	13
Activity 4.1.5	Design, Create, and Test	13
Activity 4.1.6	Evaluate and Reflect	13
Activity 4.1.7	Present Final Presentation	13