

KCKPS CTE CURRICULUM PACING GUIDE

COMPUTER SCIENCE ESSENTIALS

KSDE Course Code	41010G0.502214GGF	KSDE Course Name	Technical Introduction to Computer Science
------------------	-------------------	------------------	--

CIP Code	11.0201	KSDE Pathway	Programming & Software Development
----------	---------	--------------	------------------------------------

Infinite Campus Course Numbers	10094011S & 10094011Y
--------------------------------	-----------------------

Prerequisites	None. Some students may have taken Computer Applications in 8th Grade or 9th Grade.
---------------	--

Courses That Follow	AP Computer Science Principles, AP Computer Science A, Cyber Security, Computer Programming
---------------------	---

Buildings Offered	F. L. Schlagle, J. C. Harmon, Sumner, Washington, Wyandotte
-------------------	---

KCKPS Course Description

Computer Science Essentials introduces students to the world of computer science through project-based learning. Students will begin the semester learning to use block-based programming to develop usable apps. Then, they will transition to programming environments that display both block programming and text based programming side-by-side. Finally, students will be introduced to the Python programming language.

Kansas Department of Education Course Competencies

[Programming & Software Development Pathway](#)

This course utilizes the approved competencies provided on the KSDE website. The competencies identified by KSDE provide the foundation for what students should know and/or be able to do by the end of the course.

Common District Summative Assessment

Precision Exam: Exploring Computer Science

Description: Exploring Computer Science is designed to introduce students to the breadth of the field of computer science through an exploration of engaging and accessible topics. Rather than focusing the entire course on learning software tools or programming languages, the course is designed to focus the conceptual ideas of computing and help students understand why certain tools or languages might be utilized to solve particular problems. The goal of Exploring Computer Science is to develop in students the computational thinking practices of algorithm development, problem solving and programming within the context of problems that are relevant to the lives of today's students. Students will also be introduced to topics such as interface design, limits of computers and societal and ethical issues.

Standards: https://www.precisionexams.com/kansas/files/standards-pdfs/ks_802.pdf

PACING GUIDE AT A GLANCE

Unit	Unit Name	Length
1	Creative Computing: Building with Blocks	44 - 90-minute blocks
2	Computing and Society: Transitions to Text	33 - 90-minute blocks
3	Solving with Syntax	11 - 90-minute blocks

KSDE COURSE COMPETENCIES

Competency	Unit Taught
Design and develop a software artifact working in a team	1-3
Compare and contrast various software licensing schemes (e. g. , open source, freeware, commercial)	2
Design, develop, and implement a computing artifact that responds to an event (e. g. , robot that responds to a sensor, mobile app that responds to a text message, sprite that responds to a broadcast)	1
Use user - centered research and design techniques (e. g. , surveys, interviews) to create software solutions)	1
Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts	1-3
Deconstruct a complex problem into simpler parts using predefined constructs (e. g. , functions and parameters and/or classes)	1-3
Demonstrate the value of abstraction for managing problem complexity (e. g. , using a list instead of discrete variables)	1
Design algorithms using sequence, selection, and iteration	1-3
Use a systematic approach and debugging tools to independently debug a program (e.g. , setting breakpoints, inspecting variables with a debugger)	1-3
Convert between binary, decimal, and hexadecimal representations of data (e. g. , convert hexadecimal color codes to decimal percentages, ASCII/Unicode representation)	3
Debate the social and economic implications associated with ethical and unethical computing practices (e. g. , intellectual property rights,	1

hacktivism, software piracy, diesel emissions testing scandal, new computers shipped with malware)	
Compare and debate the positive and negative impacts of computing on behavior and culture (e. g. , evolution from hitchhiking to ridesharing apps, online accommodation rental services)	2
Compare and contrast multiple viewpoints on cybersecurity (e. g. , from the perspective of security experts, privacy advocates, the government)	1
Explain the principles of information security (confidentiality, integrity, availability) and authentication techniques	1
Use simple encryption and decryption algorithms to transmit/receive an encrypted message	1

UNIT 1

Creative Computing: Building with Blocks

44 blocks (90 minutes)

Unit Overview

Students work with MIT App Inventor to create basic apps that rely on the concepts of event-driven programming, branching, iteration, variables, and abstraction--the building blocks of creating with code. Students are introduced to essential computational thinking practices, such as developing abstractions, collaborating around computing, and communicating as they create, test, and refine computational artifacts of apps.

Unit Competencies

- Design and develop a software artifact working in a team
- Design, develop, and implement a computing artifact that responds to an event (e. g. , robot that responds to a sensor, mobile app that responds to a text message, sprite that responds to a broadcast)
- Use user - centered research and design techniques (e. g. , surveys, interviews) to create software solutions)
- Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts
- Deconstruct a complex problem into simpler parts using predefined constructs (e. g. , functions and parameters and/or classes)
- Design algorithms using sequence, selection, and iteration
- Demonstrate the value of abstraction for managing problem complexity (e. g. , using a list instead of discrete variables)
- Use a systematic approach and debugging tools to independently debug a program (e.g. , setting breakpoints, inspecting variables with a debugger)
- Debate the social and economic implications associated with ethical and unethical computing practices (e. g. , intellectual property rights, hacktivism, software piracy, diesel emissions testing scandal, new computers shipped with malware)
- Compare and contrast multiple viewpoints on cybersecurity (e. g. , from the perspective of security experts, privacy advocates, the government)
- Explain the principles of information security (confidentiality, integrity, availability) and authentication techniques
- Use simple encryption and decryption algorithms to transmit/receive an encrypted message

Unit Vocabulary

variables, pair programming, iterations, pseudo code, algorithms, abstractions, loops, database

Unit Resources

Primary Resources

- Project Lead the Way Curriculum

Supplemental Resources

-

UNIT 2

Computing and Society: Transitions to Text

33 blocks (90 minutes)

Unit Overview

Unit 2 continues to reinforce coding fundamentals as students are gradually introduced to text-based programming. In this unit, students will explore the impact of computer science on our society and bring coding off the screen and into the physical world. Students will learn how images can be used to make decisions in programs and explore real-world applications and innovations that will shape our future.

Unit Competencies

- Design and develop a software artifact working in a team
- Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts
- Deconstruct a complex problem into simpler parts using predefined constructs (e. g. , functions and parameters and/or classes)
- Compare and contrast various software licensing schemes (e. g. , open source, freeware, commercial)
- Design algorithms using sequence, selection, and iteration
- Use a systematic approach and debugging tools to independently debug a program (e.g. , setting breakpoints, inspecting variables with a debugger)
- Compare and debate the positive and negative impacts of computing on behavior and culture (e. g. , evolution from hitchhiking to ridesharing apps, online accommodation rental services)

Unit Vocabulary

one dimensional and two dimensional arrays, conditional event driven programming, data types, ethics, machine vision

Unit Resources

Primary Resources

- Project Lead the Way Curriculum

Supplemental Resources

-

UNIT 3

Solving with Syntax

11 blocks (90 minutes)

Unit Overview

The goal of Unit 3 is for students to begin to understand and use the flexibility and power of programming in a text-based environment. Students will be introduced to *Python* programming language in the collaborative Cloud9 development environment. In this unit, students will continue to build on coding fundamentals as they apply the same coding concepts, computational thinking practices, and development processes introduced in units 1 and 2.

Unit Competencies

- Design and develop a software artifact working in a team
- Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts
- Deconstruct a complex problem into simpler parts using predefined constructs (e. g. , functions and parameters and/or classes)
- Design algorithms using sequence, selection, and iteration
- Use a systematic approach and debugging tools to independently debug a program (e.g. , setting breakpoints, inspecting variables with a debugger)
- Convert between binary, decimal, and hexadecimal representations of data (e. g. , convert hexadecimal color codes to decimal percentages, ASCII/Unicode representation)

Unit Vocabulary

cloud computing, file management, procedures, tracer bullet development, class, objects

Unit Resources

Primary Resources

- Project Lead the Way Curriculum

Supplemental Resources

-