Chino Valley Unified School District
High School Course Description

CONTACTS	
1. School/District Information:	School/District: Chino Valley Unified School District Street Address: 5130 Riverside Dr., Chino, CA 91710 Phone: (909) 628-1201 Web Site: chino.k12.ca.us
2. Course Contact:	Teacher Contact: Office of Secondary Curriculum and Instruction Position/Title: Director of Secondary Curriculum and Instruction Site: District Office Phone: (909) 628-1201 X1630
A. COVER PAGE - COURSE ID	
1. Course Title:	Computer Science Essentials
2. Transcript Title/Abbreviation:	CSE
3. Transcript Course Code/Number:	5E46
4. Seeking Honors Distinction:	No
5. Subject Area/Category:	Meets the third year UC/CSU "D" Science requirement
6. Grade level(s):	9-10
7. Unit Value:	5 credits per semester/10 credits
8. Course previously approved by UC:	No
9. Course classified as a Career Technical Education course:	Yes
10. Course modeled after an UC-approved course:	Yes
11. Repeatable for credit:	No
12. Date of Board Approval:	June 17, 2021
13. Brief Course Description: Computer Science Essentials (CSE) is a yearlong course designed as an entry point for new high school computer science learners. Additionally, for students who have prior computer science experience, the course offers many opportunities for them to build upon their knowledge and skills. All students who take Computer Science Essentials will have many opportunities for creative expression and exploration in topics of personal interest, whether it be through app development, web design, or connecting computing with the physical world.	
14. Prerequisites:	None
15. Context for Course: Computer Science Essentials (CSE) is intended as a first-year introductory course for students with no prior experience with high-school level computer science courses or those that want to strengthen skills for Advanced Placement computer science courses. Computer Science Essentials introduces students to coding fundamentals through approachable, block-based programming languages where they will have early success in creating usable apps. As students sharpen their computational thinking skills, they will transition to programming environments that reinforce coding fundamentals by displaying block programming and text-based programming side-by-side. Finally, students will learn the power of text-based programming as they are introduced to the Python® programming language used in Advanced Placement Computer Science Principles.	
16. History of Course Development: This course utilizes curriculum from the Project Lead the Way (PLTW) Computer Science sequence and was developed by PLTW. Computer Science Essentials is designed with strong connections to the Computer Science K12 Frameworks (CS K12), the Computer Science Teachers Association K-12 Computer Science (CSTA K-12 CS) Level 3A Standards, integrates the CTE anchor standards and Information and communication standards, as well as the Advanced Placement Computer Science Principles (AP CSP) Frameworks. CSE qualifies as an introductory CTE course within the Information	

and Communication Technologies Sector with emphasis on the Software and Systems Development Pathway.	
17. Textbooks:	None
18. Supplemental Instructional Materials:	Access to computers with appropriate software and computer lab with appropriate tools.

B. COURSE CONTENT

1. Course Purpose:

The course engages students in computational thinking practices and collaboration strategies, as well as industry-standard tools authentic to how computer science professionals work. As a CTE introductory level course, students will learn about professional opportunities in computer science and how computing can be an integral part of all careers today. The following is a list of the units of study in the course.

2. Course Outline:

Unit 1: Creative Computing: Building with Blocks

Anchor Standards: 1.0

ICT: C1.0, C1.1, C1.2, C1.3, C1.4, C1.5, C1.6, C5.0, C5.1, C5.2, C5.3, C5.4, C5.5

This unit introduces new and returning students to the world of computer science and coding fundamentals.

- Students will 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.
- Students will create, test, and refine computational artifacts of Android™ apps.

Unit 2: Computing and Society: Transitions to Text

Anchor Standards: 3.0, 6.0, 7.0, 8.0

ICT: C2.0, C2.1, C2.2, C2.3, C2.4, C2.5, C3.0, C3.1, C3.2, C8.0, C8.1, C8.2, C8.3, C8.4, C8.5, C8.6, C8.7, C8.8

This unit reinforces coding fundamentals as students are gradually introduced to text-based programming.

- Students will explore the impacts of computer science on our society.
- Students will bring coding off the screen and into the physical world.
- Students will learn how images can be used to make decisions in programs.
- Students will explore real-world applications and innovations that will shape our future.

Unit 3: Web Development: Solving with Syntax

Anchor Standards: 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 4.0, 4.1, 4.2, 4.3, 4.4, 4.5, 4.6

ICT: C4.0, C4.1, C4.2, C4.3, C4.4, C4.5, C4.6, C4.7, C4.8, C4.9, C4.10, C4.11, C5.0, C5.1, C5.2, C5.3, C5.4, C5.5, C5.6, C6.0, C6.1, C6.2, C6.3, C6.4, C6.5, C6.6, C6.7, C7.0, C7.1, C7.2, C7.3, C7.4, C7.5, C7.6

This unit is for students to begin to understand and use the flexibility and power of programming in a text-based environment.

- Students will learn how client-side and server-side connections make the Web work.
- Students will be introduced to the Python® programming language in the collaborative Cloud9 development environment.
- Students will continue to build on coding fundamentals.
- Students will apply the same coding concepts, computational thinking practices, and development processes introduced in units 1 and 2.

Unit 4: Computing with a Purpose

Anchor Standards: 5.0, 5.1, 5.2, 5.3, 5.4, 9.0, 10.0, 10.1, 10.2, 10.3, 10.4, 10.5, 10.6

ICT: C1.0, C1.1, C1.2, C1.3, C1.4, C1.5, C1.6, C2.0, C2.1, C2.2, C2.3, C2.4, C2.5, C3.0, C3.1, C3.2, C3.3, C4.0, C4.1, C4.2, C4.3, C4.4, C4.5, C4.6, C4.7, C4.8, C4.9, C4.10, C4.11, C5.0.C5.1, C5.2, C5.3, C5.4, C5.5, C9.0, C9.1, C9.2, C9.3, C9.4

The final unit in CS Essentials allows students to apply all that they have learned in a student-defined, student- driven development.

- Students will apply computational thinking practices when creating an app, a website, or a physical computing device.
- Students will apply strategic development process to create computational artifacts that solve problems and create value for others.
- Students will collaborate the way computing professionals do as they pursue solutions to authentic needs.
- Students will model how to participate in, document, and create a performance task that provides the foundation for Advanced Placement Computer Science Principles.

Anchor Standards

1.0 Academics: Analyze and apply appropriate academic standards required for successful industry sector pathway completion leading to postsecondary education and employment. Refer to the Information and Communication Technologies academic alignment matrix for identification of standards.

2.0 Communications: Acquire and accurately use Information and Communication Technologies sector terminology and protocols at the career and college readiness level for communicating effectively in oral, written, and multimedia formats.

- 2.1 Recognize the elements of communication using a sender–receiver model.
- 2.2 Identify barriers to accurate and appropriate communication.
- 2.3 Interpret verbal and nonverbal communications and respond appropriately.
- 2.4 Demonstrate elements of written and electronic communication such as accurate spelling, grammar, and format.
- 2.5 Communicate information and ideas effectively to multiple audiences using a variety of media and formats.
- 2.6 Advocate and practice safe, legal, and responsible use of digital media information and communications technologies.
- 2.7 Use technical writing and communication skills to work effectively with diverse groups of people.
- 2.8 Understand the principles of a customer-oriented service approach to users.

3.0 Career Planning and Management: Integrate multiple sources of career information from diverse formats to make informed career decisions, solve problems, and manage personal career plans.

4.0 Technology: Use existing and emerging technology, to investigate, research, and produce products and services, including new information, as required in the Information and Communication Technologies sector workplace environment.

- 4.1 Use electronic reference materials to gather information and produce products and services.
- 4.2 Employ technology-based communications responsibly and effectively to explore complex systems and issues.
- 4.3 Use information and communication technologies to synthesize, summarize, compare, and contrast information from multiple sources.
- 4.4 Discern the quality and value of information collected using digital technologies and recognize bias and intent of the associated sources.
- 4.5 Research past, present, and projected technological advances as they impact a particular pathway.
- 4.6 Assess the value of various information and communication technologies to interact with constituent populations as part of a search of the current literature or in relation to the information task.

5.0 Problem Solving and Critical Thinking: Conduct short, as well as more sustained, research to create alternative solutions to answer a question or solve a problem unique to the Information and Communication Technologies sector using critical and creative thinking, logical reasoning, analysis, inquiry, and problem-solving techniques.

5.1 Identify and ask significant questions that clarify various points of view to solve problems.
5.2 Solve predictable and unpredictable work-related problems using various types of reasoning (inductive, deductive) as appropriate.
5.3 Use systems thinking to analyze how various components interact with each other to produce outcomes in a complex work environment.
5.4 Interpret information and draw conclusions, based on the best analysis, to make informed decisions.
5.5 Use a logical and structured approach to isolate and identify the source of problems and to resolve problems.
5.6 Know the available resources for identifying and resolving problems.
5.7 Work out problems iteratively and recursively.
5.8 Create and use algorithms and solve problems.
5.9 Deconstruct large problems into components to solve.
5.10 Use multiple layers of abstraction.
5.11 Understand the concept of base systems, including binary and hexadecimal.
5.12 Apply the concepts of Boolean logic to decision making and searching.

6.0 Health and safety: Demonstrate health and safety procedures, regulations, and personal health practices and determine the meaning of symbols, key terms, and domain-specific words and phrases as related to the Information and Communication Technologies sector workplace environment.

7.0 Responsibility and Flexibility: Initiate, and participate in, a range of collaborations demonstrating behaviors that reflect personal and professional responsibility, flexibility, and respect in the Information and Communication Technologies sector workplace environment and community settings.

8.0 Ethics and Legal Responsibilities: Practice professional, ethical, and legal behavior, responding thoughtfully to diverse perspectives and resolving contradictions when possible, consistent with applicable laws, regulations, and organizational norms.

9.0 Leadership and Teamwork: Work with peers to promote divergent and creative perspectives, effective leadership, group dynamics, team and individual decision making, benefits of workforce diversity, and conflict resolution such as those practiced in the Future Business Leaders of America and SkillsUSA career technical student organization.

10.0 Technical Knowledge and Skills: Apply essential technical knowledge and skills common to all pathways in the Information and Communication Technologies sector, following procedures when carrying out experiments or performing technical tasks.
10.1 Interpret and explain terminology and practices specific to the Information and Communication Technologies sector.
10.2 Comply with the rules, regulations, and expectations of all aspects of the Information and Communication Technologies sector.
10.3 Construct projects and products specific to the Information and Communication Technologies sector requirements and expectations.
10.4 Collaborate with industry experts for specific technical knowledge and skills.
10.5 Understand the major software and hardware components of a computer and a network and how they relate to each other.
10.6 Understand data sizes of various types of information (text, pictures, sound, video, etc.) and data capacity of various forms of media.

Information and Communication Technologies Pathway Standards
Software and Systems Development Pathway

C1.0 Identify and apply the systems development process.
- C1.1 Identify the phases of the systems development life cycle, including analysis, design, programming, testing, implementation, maintenance, and improvement.
- C1.2 Identify and describe models of systems development, systems development life cycle (SDLC), and agile computing.
- C1.3 Identify and describe how specifications and requirements are developed for new and existing software applications.
- C1.4 Work as a member of, and within the scope and boundaries of, a development project team.
- C1.5 Track development project milestones using the concept of versions.
- C1.6 Diagram processes using flowcharts and the Unified Modeling Language.

C2.0 Define and analyze systems and software requirements.
- C2.1 Describe the major purposes and benefits of development, including automation, improving productivity, modeling and analysis, and entertainment.
- C2.2 Recognize and prevent unintended consequences of development work: programming errors, security issues, health and environmental risks, and privacy concerns.
- C2.3 Develop strategies that target the specific needs and desires of the customer.
- C2.4 Analyze customers' needs for development.
- C2.5 Determine and document the requirements and alternative solutions to fulfill the customers' needs.

C3.0 Create effective interfaces between humans and technology.
- C3.1 Describe and apply the basic process of input, processing, and output.
- C3.2 Design effective and intuitive interfaces using knowledge of cognitive, physical, and social interactions.

C4.0 Develop software using programming languages.
- C4.1 Identify and describe the abstraction level of programming languages from low-level, hardware-based languages to high-level, interpreted, Web-based languages.
- C4.2 Describe the interaction and integration of programming languages and protocols such as how client-side programming can work with server-side programming to use a query language to access a database.
- C4.3 Identify and use different authoring tools and integrated development environments (IDEs).
- C4.4 Identify and apply data types and encoding.
- C4.5 Demonstrate awareness of various programming paradigms, including procedural, object oriented, event-driven, and multithreaded programing.
- C4.6 Use proper programming language syntax.
- C4.7 Use various data structures, arrays, objects, files, and databases.
- C4.8 Use object-oriented programming concepts, properties, methods, and inheritance.
- C4.9 Create programs using control structures, procedures, functions, parameters, variables, error recovery, and recursion.
- C4.10 Create and know the comparative advantages of various queue, sorting, and searching algorithms.
- C4.11 Document development work for various audiences, such as comments for other programmers, and manuals for users.

C5.0 Test, debug, and improve software development work.
- C5.1 Identify the characteristics of reliable, effective, and efficient products.
- C5.2 Describe the ways in which specification changes and technological advances can require the modification of programs.
- C5.3 Use strategies to optimize code for improved performance.

C5.4 Test software and projects.C5.5 Evaluate results against initial requirements.
C5.6 Debug software as part of the quality assurance process.

C6.0 Integrate a variety of media into development projects.
C6.1 Identify the basic design elements necessary to produce effective print, video, audio, and interactive media.
C6.2 Describe the various encoding methods of media and trade-offs: vector graphics vs. bitmaps, and bit depth.
C6.3 Use media design and editing software: keyframe animation, drawing software, image editors, and three-dimensional design.
C6.4 Develop a presentation or other multimedia project: video, game, or interactive Web sites, from storyboard to production.
C6.5 Analyze the use of media to determine the appropriate file format and level of compression.
C6.6 Integrate media into a full project using appropriate tools.
C6.7 Create and/or capture professional-quality media, images, documents, audio, and video clips.

C7.0 Develop Web and online projects.
C7.1 Identify the hardware (server) and software required for Web hosting and other services.
C7.2 Describe the full process of online content delivery, registering domain names, setting up hosting, and setting up e-mail addresses.
C7.3 Attract Web-site visitors through search engine optimization using various strategies like keywords and meta-tags.
C7.4 Enable e-commerce capabilities to sell products, create a shopping cart, and handle credit card transactions.
C7.5 Create an online project, Web-based business, and e-portfolio.
C7.6 Optimize fast delivery and retrieval of online content such as Web pages.

C8.0 Develop databases.
C8.1 Describe the critical function of databases in modern organizations.
C8.2 Identify and use the basic structures of databases, fields, records, tables, and views.
C8.3 Identify and explain the types of relationships between tables (one-to-one, one-to-many, many-to-many) and use methods to establish these relationships, including primary keys, foreign keys, and indexes.
C8.4 Use data modeling techniques to create databases based upon business needs.
C8.5 Use queries to extract and manipulate data (select queries, action queries).
C8.6 Develop databases that are properly normalized using appropriate schemas.
C8.7 Export and import data to and from other applications and a database recognizing the limitations and challenges inherent in the process.
C8.8 Analyze and display data to assist with decision making using methods like cross tabulations, graphs, and charts.

C9.0 Develop software for a variety of devices, including robotics.
C9.1 Demonstrate awareness of the applications of device development work, including personalized computing, robotics, and smart appliances.
C9.2 Install equipment, assemble hardware, and perform tests using appropriate tools and technology.
C9.3 Use hardware to gain input, process information, and take action.
C9.4 Apply the concepts of embedded programming, including digital logic, machine-level representation of data, and memory-system organization.

Key Assignments:

App Development: Creating Value forOthers

Student groups will apply development strategies and user-centered research to create an app that has value to others. Students will gain insight on the importance of creativity, persistence, and value of diverse perspectives in an iterative

development process.

CSTA Standards: 3A-A-2-1, 3A-A-5-5, 3A-A-5-6, 3A-A-6-12, and 3A-I-2-22,

- Students will design and develop a software artifact working in a team.
- Students will use user-centered research and design techniques (e.g., surveys, interviews) to create software solutions.
- Students will integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
- Students will use a systematic approach and debugging tools to independently debug a program (e.g., setting breakpoints, inspecting variables with a debugger).
- Students will 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).

Image Processing: Cooperative Driving and Self-driving Vehicles

Tomorrow's solutions involve all of us. In the final lesson, student groups will learn how to take collaborations to scale to achieve a common goal by designing solutions to image processing algorithms with application in AI and self-driving cars.

CSTA Standards: 3A-A-5-6, 3A-A-4-8, and 3A-A-4-9

- Students will integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
- Students will reconstruct a complex problem into simpler parts using predefined constructs (e.g., functions and parameters and/or classes).
- Students will demonstrate the value of abstraction for managing problem complexity (e.g., using a list instead of discrete variables).

Game Simulation Project: Using Coding Constructs

Students create a game simulation and reinforce what they have learned 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 career fields.

CSTA Standards: 3A-A-5-4, 3A-A-5-6, 3A-A-3-10, 3A-C-5-14, and 3A-D-4-18

- Students will 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).
- Students will integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
- Students will design algorithms using sequence, selection, and iteration.
- Students will create, extend, or modify existing programs to add new features and behaviors using different forms of inputs and outputs (e.g., inputs such as sensors, mouse clicks, data sets; outputs such as text, graphics, sounds).
- Students will convert between binary, decimal, and hexadecimal representations of data (e.g., convert hexadecimal color codes to decimal percentages, ASCII/Unicode representation).

Web Development: Creating Your Own Website

This lesson will allow students to collaboratively design, create, and connect a secure website based on an interest or need that the student group defines. Student groups will apply development strategies and user-centered research to create a website that has value to others and protects sensitive data.

CSTA Standards: 3A-D-3-20, 3A-I-6-29, 3A-N-7-30, 3A-N-1-32, and 3A-N-3-34

- Students will discuss techniques used to store, process, and retrieve different amounts of information (e.g., files, databases, data warehouses).

- Students will redesign user interfaces (e.g., web pages, mobile applications, animations) to be more inclusive, accessible, and minimizing the impact of the designer's inherent bias.
- Students will describe key protocols and underlying processes of Internet-based services (e.g., http/https and SMTP/IMAP, routing protocols).
- Students will compare and contrast multiple viewpoints on cybersecurity (e.g., from the perspective of security experts, privacy advocates, the government).
- Students will use simple encryption and decryption algorithms to transmit/receive an encrypted message.

Class Capstone Task: Develop an Artifact Important to You

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.

CSTA Standards: 3A-A-2-1, 3A-A-2-2, 3A-A-5-6, 3A-I-2-22, 3A-I-1-27, and 3A-I-6-29

- Students will design and develop a software artifact working in a team.
- Students will demonstrate how diverse collaborating impacts the design and development of software products (e.g., discussing real-world examples of products that have been improved through having a diverse design team or reflecting on their own team's development experience).
- Students will integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.
- Students will 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).
- Students will demonstrate how computing enables new forms of experience, expression, communication, and collaborating.
- Students will redesign user interfaces (e.g., webpages, mobile applications, animations) to be more inclusive, accessible, and minimizing the impact of the designer's inherent bias.

Computer Science Teachers Association (CSTA) Standards

3A-A-2-1 Computer Science Teachers Association (CSTA): Design and develop a software artifact working in a team.

3A-A-2-2 Computer Science Teachers Association (CSTA): Demonstrate how diverse collaborating impacts the design and development of software products (e.g., discussing real-world examples of products that have been improved through having a diverse design team or reflecting on their own team's development experience).

3A-A-3-10 Computer Science Teachers Association (CSTA) Design algorithms using sequence, selection, and iteration.

3A-A-4-8 Computer Science Teachers Association (CSTA): Deconstruct a complex problem into simpler parts using predefined constructs (e.g., functions and parameters and/or classes).

3A-A-4-9 Computer Science Teachers Association (CSTA): Demonstrate the value of abstraction for managing problem complexity (e.g., using a list instead of discrete variables).

3A-A-5-4 Computer Science Teachers Association (CSTA): 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).

3A-A-5-5 Computer Science Teachers Association (CSTA): Use user-centered research and design techniques (e.g., surveys, interviews) to create software solutions.

3A-A-5-6 Computer Science Teachers Association (CSTA): Integrate grade-level appropriate mathematical techniques, concepts, and processes in the creation of computing artifacts.

3A-A-6-12 Computer Science Teachers Association (CSTA): Use a systematic approach and debugging tools to independently debug a program (e.g., setting breakpoints, inspecting variables with a debugger).

3A-C-5-14 Computer Science Teachers Association (CSTA): Create, extend, or modify existing programs to add new features and behaviors using different forms of inputs and outputs (e.g., inputs such as sensors, mouse clicks,

data sets; outputs such as text, graphics, sounds).

3A-D-3-20 Computer Science Teachers Association (CSTA): Discuss techniques used to store, process, and retrieve different amounts of information (e.g., files, databases, data warehouses).

3A-D-4-18 Computer Science Teachers Association (CSTA): Convert between binary, decimal, and hexadecimal representations of data (e.g., convert hexadecimal color codes to decimal percentages, ASCII/Unicode representation).

3A-I-1-27 Computer Science Teachers Association (CSTA): Demonstrate how computing enables new forms of experience, expression, communication, and collaborating.

3A-I-2-22 Computer Science Teachers Association (CSTA): 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).

3A-I-6-29 Computer Science Teachers Association (CSTA): Redesign user interfaces (e.g., webpages, mobile applications, animations) to be more inclusive, accessible, and minimizing the impact of the designer's inherent bias.

3A-N-1-32 Computer Science Teachers Association (CSTA): Compare and contrast multiple viewpoints on cybersecurity (e.g., from the perspective of security experts, privacy advocates, the government).

3A-N-3-34 Computer Science Teachers Association (CSTA): Use simple encryption and decryption algorithms to transmit/receive an encrypted message.

3A-N-7-30 Computer Science Teachers Association (CSTA): Describe key protocols and underlying processes of Internet-based services (e.g., http/https and SMTP/IMAP, routing protocols).

4. Instructional Methods and/or Strategies:

APB (Activity, Project, and Problem-based) Instructional Design providing students with unique opportunities to work collaboratively, identify problems, apply what they know, persevere through challenges, find unique solutions, and lead their own learning.

5. Assessment Including Methods and/or Tools:

- Project-based assessments using APB rubrics.
- A computer-based End of Course (EOC) exam delivered online.
- LMS system supports delivery of curriculum and assessments.

The evaluation of student progress and evaluation will be based on the following criteria outlined in Board Policy:

- Assessments: 60-75% of the final grade
- Assignments and class discussions: 25-40% of the final grade