Chino Valley Unified School District
High School Course Description

Cal	
1. School/District Information:	School/District: Chino Valley Unified School District Street Address: 5130 Riverside Dr. Phone: (909) 628-1201 Web Site: chino.k12.ca.us
2. Course Contact:	Teacher Contact: Office of Secondary Curriculum Position/Title: Director of Secondary Curriculum Site: District Office Phone: (909) 628-1201 X1630
B. COVER PAGE - COURSE ID	
1. Course Title:	Integrated Mathematics 1 with Computing and Robotics
2. Transcript Title/Abbreviation:	IM1 CRP
3. Transcript Course Code/Number:	5M01
4. Seeking Honors Distinction:	No
5. Subject Area/Category:	Meets UC/CSU “c” Mathematics requirement
6. Grade Level(s):	9-12
7. Unit Value:	5 credits per semester/ 10 credits total
8. Course Previously Approved by UC:	No
9. Classified as a Career Technical Education Course:	Yes
10. Modeled after an UC-approved course:	Yes
11. Repeatable for Credit:	No
12. Date of Board Approval:	May 16, 2019

13. Brief Course Description:
Integrated Mathematics 1 with Computing and Robotics prepares students for both college and career by integrating computing and robotics technologies with mathematics instruction. The course guides students through topics in Integrated Mathematics 1 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. The course is aligned with Career Technical Education (CTE) standards in the Information and Communications Technology (ICT) industry sector.

14. Prerequisites:	

15. Context for Course: Students use programming in C/C++ interpreter Ch, a proprietary cross-platform scripting and language environment to reinforce and extend their knowledge of mathematical concepts by analyzing real life situations, identifying given information, formulating steps that a computer program could calculate to find a solution, analyzing the results for accuracy, and revising/modifying the programming solutions as necessary. Robotics activities allow students to reenact physically derived mathematical problems through robotics technologies, such as RoboBlockly and RoboSim via C-STEM Studio, to visualize situations, associate linear and exponential graphs with physical phenomenon, predict and identify key features of the graphs with robotic systems, and solve robotics problems through mathematical modeling and programming.

16. History of Course Development:
The course was designed to provide students with the skills and knowledge to be successful in successive Integrated Mathematics courses as well as in the Software and Systems Development career pathway.

17. Textbooks:	Cheng, H. H. (2016). *Learning Common Core Mathematics with C/C++ Interpreter Ch* (2nd ed.). CA: UC Davis C-STEM Center.
18. Supplemental Instructional Materials:	Cheng, H.H. (2016) *Learning Computer Programming with Ch for the Absolute Beginner.* (6th ed.) CA: UC Davis C-STEM Center.

	Cheng, H.H. (2018) *Learning Robot Programming with Linkbot for the Absolute Beginner.* (7th ed.) CA: UC Davis C-STEM Center. C/C++ Software

C. COURSE CONTENT

1. Course Purpose:

This course is designed for the California Career and Technical Education Information and Communication Technologies (ICT) sector. This course is aligned to the California Career and Technical Education Standards: Software and Systems Development pathway and is designed to be a concentrator level course.

2. Course Outline:

Unit 1: Introduction to Computing and Robotics/Operations with Real Numbers and Expressions

This unit introduces students to how a computer works and the importance of computing in the 21st century.

- Students will learn the basics of programming and programming language syntax in C/C++ using the C/C++ interpreter Ch.
- Students will evaluate expressions and practice order of operations in a Ch Command Shell.
- Students will write proper programming language syntax to review and practice basic operations with real numbers, order of operations, and manipulating and evaluating variables in simple algebraic equations.
- Students will persevere in solving specific problems with attention to precision, construct variable arguments and critique the reasoning of others, and model with mathematics.
- Students will understand problems that arise in real life context of programming with robotics and find solutions of multi-step problems, choose and interpret the problems with formulas and conceptual understanding, and choose and interpret the scale of measurement.
- Students will apply their knowledge and understanding of basic programming syntax and number sense, expressions and equations to create mathematical models.

Unit 2: Using Functions and Robotics for Math Application

In this unit, students learn function notations and develop the concepts of domain and range.

- Students will explore four types of functions (arithmetic sequence, geometric sequence, linear, and exponential) and interpret them graphically, numerically, symbolically, and verbally.
- Students will interpret arithmetic sequences and geometric sequences as the linear functions and exponential functions.
- Students will work as a development project team to construct programs in Ch that define a function, call the function using correct syntax, and debug it.
- Students will construct graphs of functions using **plot.func2D() and plot.expr()** with arguments specific to the graphed function.
- Students will graph linear and exponential function with transformations.
- Students will diagram processes using flowcharts.
- Students will integrate a variety of media into development projects, develop web and online projects, and develop programs that control the motions of robotics using **robot.driveDistance()** and **robot.drivexyToFunc()**.
- Students will practice and improve their writing, reading, listening, and language skills.

Unit 3: Linear Models and Solving Linear Equations and Inequalities

In this unit, students work on multiple tasks integrating on concepts of mathematics, software development, and robotics.

- Students connect two dimensional lines and systems of linear equations as well as inequalities algebraically and graphically using robots and programming.

- Students will learn the graph of a linear function is the set of all ordered pair of solutions plotted on a plane as well as the meaning of the solution to a system of two linear equations by using the robot simulation environment RoboBlockly and RoboSim with virtual robots.
- Robots are used to deepen the students' understanding of what a line means and the meaning of two crossing lines in terms of real-life situations. This is reinforced by graphical output of the two programs *recorddistance.ch* and *recorddistanceoffset.ch.*
- Students learn to plot all solutions to any equation in two variables, graph the equations while displaying its data, and run the code to explore what happens when one changes the speed or distance that a robot travels.
- Students will make sense of problems and persevere in solving them, reason abstractly and quantitatively, and attending to precision by focusing on Standards of Mathematical Practice.

Unit 4: Statistical Data Analysis

In this unit, students learn to reason abstractly and quantitatively to create plots with a title, labels, and specific points using member functions **plot.title(), plot.label(),** and **plot.point()**, respectively.

- Students will make use of copying, pasting, and printing the displayed plot.
- Students will informally fit a straight line to a scatter plot and find the trend line for the data.
- Students will summarize, represent and interpret data using single variable statistical measures like mode, mean, median, and standard deviation.
- Students will use simple linear regression and residuals to analyze two-variable data. The data can also be interpreted using statistical models like scatter plots, dot plots, bar graphs, histograms, and Box-and-Whisker plots.

Unit 5: Congruency and Geometric Transformation

In this unit, students apply previously gained knowledge of software development and robots to the mathematical concepts of congruence, similarity and the four primary geometric transformations of translation, rotation, and reflection by programming a pair of unconnected robots to be simultaneously moved with identical movements, except the second robot is affected by a type of transformation.

- Students will learn how geometric transformations are applied to the movement of objects in a plane through rules that define that motion.
- Students will apply RoboSim with the x and y coordinate system when writing their computer programs in Ch.
- Students expand their computer programs with plotting that visually demonstrate and reinforce the different types of primary geometric transformations and coordinate geometry.
- Students will develop an understanding of rigid motions and similarity as it applies to polygons and transformation after the introduction of the concept of congruency.
- Student will use best programming practices and apply mathematical concepts precisely.
- Students will practice using appropriate tools strategically and attend to precision.

Unit 6: Coordinate Geometry and Geometric Construction

In this unit, students build depth of knowledge upon their prior understanding of geometry with a Cartesian coordinate system to verify geometric relationships, including properties of special triangles, quadrilaterals, circles, and slopes of parallel and perpendicular lines, as well as areas and perimeters of polygons.

- Students will design computer programs for other geometric shapes including triangles and other polygons, introducing congruence and geometric transformations by noting the robot is pre-imaged and applying the unit shifts from the translated coordinates.
- Students will expand their programming capacity by adding a control and loop structure in programming, continually, building one task after another. This extends into designing three additional computer programs for a triangle with reflection transformations under three circumstances.

- Students will design additional computer programs for other geometric shapes, while taking the original coordinates for the first robot and applying different scale factors to the second robot.
- Students will use coordinates to prove simple geometric theorems algebraically; this includes the slope formula and expressing geometric properties with equations.
- Students will communicate their reasoning, demonstrate conceptual and procedural fluency, find examples of connecting Algebra and Geometry through coordinates in real life contexts by applying the elements of mathematical modeling, and become efficient problem solvers.
- Students will use a compass, straightedge, and protractor to copy a line segment, copy an angle, bisect a segment, construct a perpendicular line, bisect an angle, construct a line parallel to a given line through a point, construct a regular hexagon inscribed in a circle, construct an equilateral triangle, and construct a square.

3. Key Assignments:

Unit 1: Students can create a program that successfully makes a robot move along a number line by evaluating positive and negative integers as it relates to measurement of distance. In this task, students are introduced to the functions in Ch by declaring and initializing variables and the basic programming syntax. In order for the robot to move along the number line, students must create a number line with a scale of 1 unit integer = 1 inch. Students then create another conversion for converting distance measurements in inches to robot joint angles. The program should allow input and output functions of passing distance and displaying questions and distance traveled by the robot. The robots move along the number line using the member functions **robot.driveDistance(), robot.driveAngle(), and robot.driveTime().**

Unit 2: Students can write a program that directs a robot's motions to be based on distance, time, and different speeds to interpret functions numerically, symbolically, and verbally, Students create a table of inputs and outputs for the distance equals speed multiplied by time function with different speeds. Next, students write a program that uses the function **robot.drivexyToFunc()** to direct the robot's motions with different speeds used and prints a list of distances traveled by the robot.

Unit 3: An object moving at a constant rate is a good example of a linear equation in two variables. Students working in pairs of two run the code *recorddistance.ch*. Students explore what happens when one changes the speed or distance that the robot travels. Then, they record the distance as the robot moves with a time interval of 0.1 second. Once the students have a strong grasp of how the speed and distance affect its graph, students summarize what they learned by writing a short essay.

Unit 4: Students plot points with the x- and y- values given on a chart to determine the relationship between the points and to form a linear equation. Additionally, students need to be in groups of 2-4 with 2 robots per group. Students will create a 4 by 4 grid and place the two robots at teacher-defined (x,y) coordinates. The goal of this activity is to have the robots be aligned in a straight line with a slope of 1 when they finish moving. Students apply their understanding of linear relationships by moving the robots in a straight line.

Unit 5: To enhance students' understanding of a geometric translation, they create a computer program in Ch to make a robot move in the geometric shape of a rectangle. Students test their program first using RoboSim with the x and y coordinate system and visible tracking to illustrate a correct geometric shape. Once students have a correctly working simulation, students then take the original coordinates and apply specific shifts to a second robot. By adjusting the second robot to the translated coordinates, and re-executing the program, students observe both robots creating the exact same rectangle shape, one at the original location, and one at a different location.

Unit 6: Students can create an island on which there are many geometrically shaped buildings blocking a straight path from one end of the island to the opposite end in RoboSim. Teams in the class will design a path that is the shortest route to get from one end of the island to the other opposite end. Then, they write a program in Ch which will control a robot to complete the route they designed. Upon completion of this task, they time how long it takes the robot to complete the course. With this benchmark time, other students in the class will compete in designing a path that beats the best time needed to complete the route by a robot.

4. Instructional Methods and/or Strategies:

- Direct instruction
- Hands on labs
- Project based learning
- Work based learning
- Collaborative environment
- Modeling

5. Assessment Including Methods and/or Tools:

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