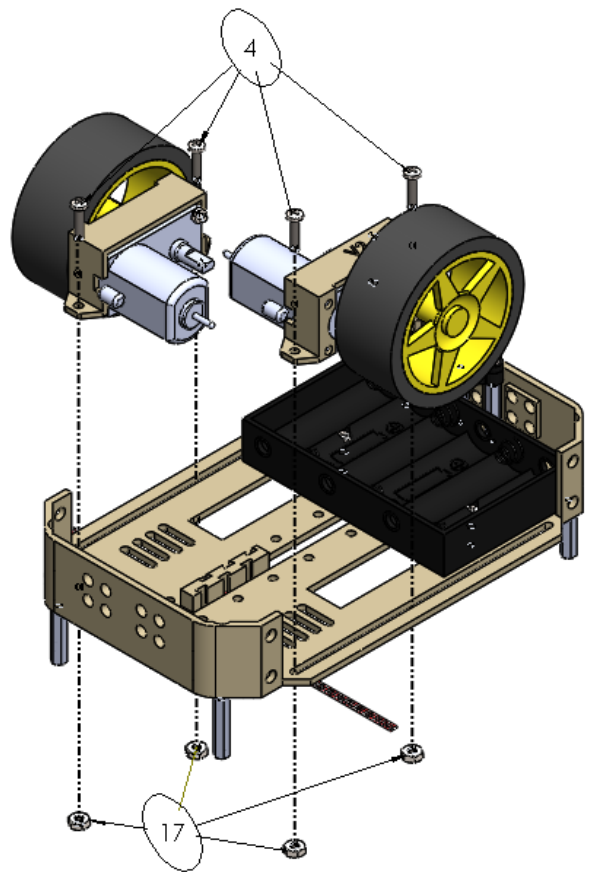
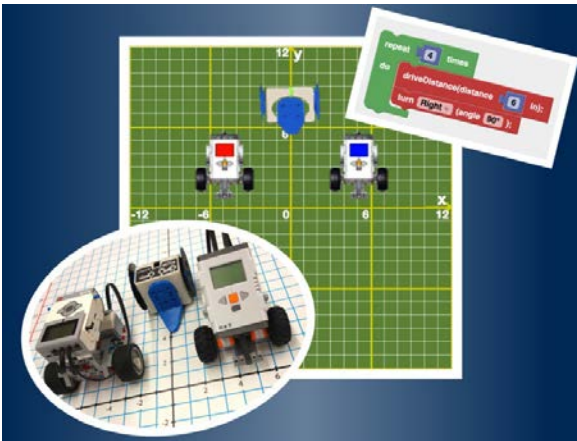


# C-STEM Math-ICT Curriculum

*13 Years Hands-on Integrated Math and Computer Science Education*



UC Davis Center for Integrated Computing and STEM Education (C-STEM)

<http://c-stem.ucdavis.edu>  
[info@c-stem.ucdavis.edu](mailto:info@c-stem.ucdavis.edu)

**Copyright © 2016-2018, UC Davis C-STEM Center, All rights reserved.** No part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written consent of the author, including, but not limited to, in any network or other electronic storage or transmission, or broadcast for distance learning.

**Acknowledgements:**

This work was supported in part by the National Science Foundation under grant numbers CNS-1132709, IIS-1208690, IIS-1256780, and by the CaMSP Program of the California Department of Education.

**UC DAVIS**  
**C-STEM Center**



## 1. Introduction

*The innovative C-STEM Math-ICT Curriculum will provide K-12 students with 13 years of hands-on integrated math and computer science education with coding in Blockly and Ch/C/C++.* The UC Davis C-STEM Center aims to transform computing, science, technology, engineering, and mathematics (C-STEM) education in both formal and informal K-14 programs through integrated learning, guided by two key objectives:

- Close the achievement gap by broadening participation of students traditionally underrepresented in computing and STEM related careers and post-secondary study.
- Develop students' 21st century problem-solving skills to tackle real world concerns through integrated computing and STEM education.

Through cutting edge research made possible with funding from the National Science Foundation and California Department of Education, the C-STEM Center, in collaboration with our industry partners, developed innovative educational technology of C-STEM Studio and RoboBlockly for K-14 students. [C-STEM Studio](#) is a platform for teaching science, technology, engineering, and mathematics (STEM) through *computing and robotics* for integrated learning. [RoboBlockly](#) is a web-based robot simulation environment. It uses a simple puzzle-piece interface to program virtual Linkbot and Lego Mindstorms NXT/EV3. The C-STEM Center works with K-14 educators to integrate computer programming and robotics into mathematics and other STEM subjects. Through the creation of C-STEM Studio and RoboBlockly, teachers are able to access project-based computing and robotics activities, integrated curriculum, and hands-on personalized and collaborative learning strategies aligned with Common Core State Standards (CCSS) and Career and Technical Education (CTE) Standards. This integration helps students make meaningful connections between STEM lessons and courses and their relevance to real-life application. C-STEM lessons are designed to develop students' critical thinking, problem-solving skills and teamwork skills. The C-STEM Studio and RoboBlockly are available to all schools at no cost.

The C-STEM program aims to provide formal computing education for all K-12 students for 13 years. The C-STEM Center studies how to use innovative computing and robotics technologies to increase student interest and help them learn STEM subjects with an emphasis on Algebra, a gatekeeper for high-school graduation, university education and careers in STEM fields. The program is designed to help close the mathematics achievement gap through engagement strategies that target traditionally unrepresented groups and at risk students. The C-STEM

Center also studies how to integrate computing education into the STEM subjects in elementary schools, middle schools, high schools, and the first two years of college, in order to positively impact student interest, persistence and retention of computing and other STEM related careers and post-secondary study.

**C-STEM is a UC Approved Educational Preparation Program** for undergraduate admission to all UC campuses. Participation in the C-STEM program, C-STEM student individual and team awards, and extracurricular activities are recognized in the UC admissions process as achievements that have explicitly prepared students for college and career.

**C-STEM has UC A-G Program Status.** High schools can readily and easily add the “A-G approved” rigorous C-STEM curriculum integrated with computing and robotics to their own school’s A-G course lists to satisfy the UC/CSU admission requirements.

## 2. C-STEM Math-ICT Curriculum

The C-STEM Math Information and Communication Technologies (ICT) Curriculum is specially designed to prepare K-12 students for Computing and STEM related careers and post-secondary studies. **The C-STEM Math-ICT Curriculum provides K-12 students with 13 years of computer science education through hands-on integrated learning of math and computer science with coding in Blockly and Ch/C/C++.** The Curriculum includes robotics and math with coding activities for elementary school students, computer programming and robotics courses for middle school students, computer programming and robotics courses, AP Computer Science Principles, and a capstone course on Principles and Design of Cyber-Physical Systems for high school students, as well as Common Core compliant math curriculum integrated with computing and robotics.

The rigorous curriculum teaches C/C++ for computing. C/C++ is the foundation for ICT and the most widely used programming language in universities and the ICT industry. Students on the C-STEM Math-ICT Curriculum are exposed to computational thinking and coding in Blockly and Ch/C/C++ for 13 years, woven together with math, science, engineering, and even artistic projects such as video production.

The C-STEM Math-ICT Curriculum uses integrated learning approaches. Students will learn math with ICT and applications, programming in Blockly and Ch/C/C++, computational thinking, algorithm development, robotics, microcontroller, design thinking, 3D modeling and 3D printing for rapid prototyping and product realization with ICT, building Cyber-Physical Systems (CPS), etc. All courses are pre-approved with A-G credit to fulfill admission requirements for the UC and California State University systems.

Building on the proven educational computing and robotics technologies, including [C-STEM Studio](#) and [RoboBlockly](#), the C-STEM Math-ICT Curriculum will allow school districts to meet President Obama’s goals of [“Computer Science for All”](#) and [“offering every student the hands-on computer science and math classes that make them job-ready on day one.”](#)

The curriculum in the 12-year C-STEM Math-ICT Curriculum can also be selectively used ranging from four weeks as supplementary materials for STEM courses, one semester course, to two semesters course in formal curriculum, multi-year career pathways or academies, as well as afterschool programs, and youth camps in both academic year and summer in school campuses and colleges.

## **2.0C-STEM Math-ICT Curriculum for Kindergarten:**

Kindergarten Math Activities in RoboBlockly **(in development)**

## **2.1 C-STEM Math-ICT Curriculum for Elementary Schools:**

### **2.1.1 Core Math Courses:**

1. Exploring Math with Computing and Robotics **(in development)**

### **2.1.2 Coding, Robotics, and Math Activities in RoboBlockly:**

2. Robotics Activities Level 1 in RoboBlockly
3. Coding Activities Level 1 in RoboBlockly
4. Grade 1 Math Activities in RoboBlockly
5. Grade 2 Math Activities in RoboBlockly
6. Grade 3 Math Activities in RoboBlockly
7. Grade 4 Math Activities in RoboBlockly
8. Grade 5 Math Activities in RoboBlockly
9. Grade 6 Math Activities in RoboBlockly

### **2.1.3 Core Science Courses:**

Science teachers can integrate C-STEM computing, robotics, design thinking, and computational thinking into all science courses.

### **2.1.4 Enhancements:**

- C-STEM Day with RoboPlay Competitions and Student Recognition.
- Film Production Projects for RoboPlay Video Competition.
- Afterschool Robotics Clubs.

## 2.2 C-STEM Math-ICT Curriculum for Middle Schools:

### 2.3.1 Core Math Courses:

1. Math 7 with Computing
2. Math 8 with Computing

### 2.3.2 Core Science Courses:

Science teachers can integrate C-STEM computing, robotics, design thinking, and computational thinking into all science courses.

### 2.3.3 ICT Elective Courses:

1. Robotics and Film Production
2. Introduction to Computer Programming
3. Introduction to Physical Computing and Making

### 2.3.4 Coding, Robotics, and Math Activities in RoboBlockly:

1. Robotics Activities Level 2 in RoboBlockly
2. Coding Activities Level 2 in RoboBlockly
3. Grade 7 Math Activities in RoboBlockly
4. Grade 8 Math Activities in RoboBlockly
5. Projects in RoboBlockly

### 2.3.5 Enhancements:

- C-STEM Day with RoboPlay Competitions and Student Recognition.
- Film Production Projects for RoboPlay Video Competition.
- Afterschool Robotics Clubs.
- Participating in [Girls in Robotics Leadership \(GIRL\) Camps](#)

## 2.3 C-STEM Math-ICT Curriculum for High Schools:

### 2.3.1a Core Math Courses in Integrated Path

1. Integrated Math 1 with Computing and Robotics
2. Integrated Math 2 with Computing and Robotics **(in development)**
3. Integrated Math 3 with Computing and Robotics **(in development)**

### 2.3.1b Core Math Courses in Traditional Path

1. Algebra 1 with Computing and Robotics
2. Geometry with Computing and Robotics **(in development)**
3. Algebra 2 with Computing and Robotics **(in development)**

### **2.3.2 Core Science Courses:**

Science teachers can integrate C-STEM computing, robotics, design thinking, and computational thinking into all science courses.

### **2.3.3 ICT Elective Courses:**

4. Computer Programming for Solving Applied Problems
5. Computing with Robotics
6. Physical Computing with Pi and Arduino
7. AP Computer Science Principles **(pilot phase)**
8. Principles and Design of Cyber-Physical Systems (the C-STEM Math-ICT Curriculum capstone course, **pilot phase**)

### **2.3.4 Coding, Robotics, and Math Activities in RoboBlockly:**

1. Robotics Activities Level 2 in RoboBlockly
2. Coding Activities Level 2 in RoboBlockly
3. Algebra Activities in RoboBlockly
4. Projects in RoboBlockly

### **2.3.5 Enhancements:**

- C-STEM Day with RoboPlay Competitions and Student Recognition.
- Film Production Projects for RoboPlay Video Competition.
- Afterschool Robotics Clubs.
- Serving as Assistant Coaches for Girls in Robotics Leadership (GIRL) Camps
- Participating in [Girls in Robotics Leadership+ \(GIRL+\)](#) Camps

## **3. Professional Development for Teaching C-STEM Math-ICT Curriculum Courses**

To ensure successful implementation of the C-STEM program, the C-STEM Center provides [unique C-STEM professional development](#) for STEM teachers, including those who have no prior computer programming and robotics experience, to implement the C-STEM Math-ICT Curriculum effectively.

The C-STEM Center can bring the C-STEM professional development opportunities to your school, district, county, and region through the following [C-STEM professional development](#).

- **1-Day Workshop on Physical Computing with Arduino and Robots**
- **2-Day Workshop on Integrated Computing and STEM Education with Raspberry Pi, Arduino, and Robots**
- **2-Day Academy on Integrated Computing and STEM Education**
- **1-Week Institute on Integrated Computing and STEM Education**
- **1-Week Workshop on Arduino, Raspberry Pi, and Cyber-Physical Systems**
- **On-Site Training**
- **Train-the-Trainer Affiliate Program**

## **4 Catalog Information for Elementary School Courses for the C-STEM Math-ICT Curriculum**

The course outlines and materials needed for implementation are available at:  
<http://c-stem.ucdavis.edu/curriculum/c-stem-elementary-school-curriculum/>

### **4.1 Exploring Mathematics with Computing and Robotics**

This course explores mathematical concepts in the Common Core State Standards-Mathematics through practical applications with hands-on and fun computing and robotics activities. Students write C/C++ computer programs to control a single robot and multiple robots. Through both personalized and collaborative group computing and robotics activities, students learn and reinforce the algebraic thinking with arithmetic operations in whole and decimal numbers, number line, fractions, measurement, variables, data conversion, lines, angles, ratios, proportions, and linear relation. The hands-on computing and experiments help students make meaningful connections between abstract math concepts and their relevance to real-life applications, as well as help develop students' critical thinking and problem-solving skills.

## **5 Catalog Information for Middle School Courses for the C-STEM Math-ICT Curriculum**

The course outlines and materials needed for implementation are available at:  
<http://c-stem.ucdavis.edu/curriculum/middle-school/>

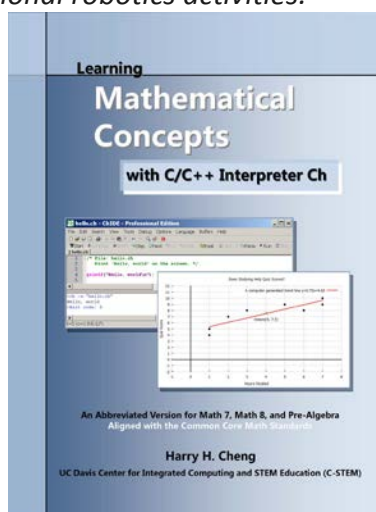
### **5.1 Math 7 with Computing**

This course, based on the Common Core Math 7 standards, uses computing to develop and expand students' understanding of Math 7 topics. Students analyze real life situations, identify given information, formulate mathematical steps to find a solution, and analyze the results for accuracy, all within the context of computer programming. The logical process of computer programming allows students to organize their approach to problem solving and efficiently



analyze and correct their work. Topics covered include evaluating expressions, one variable equations and inequalities, rates, proportions, percents, probability, similarity, plotting points and linear equations, and identifying slopes and intercepts. Optional group computing activities allow students to collaborate on critical thinking activities based on algebraic topics while developing their ability to effectively communicate, listen, share responsibility and respectfully address the suggestions of others. Optional robotics extension activities allow students to reenact physically derived mathematical problems through robotics technologies to visualize situations, associate graphs with physical phenomenon, predict and identify key features of the graphs with the specific physical situations, and solve physical problems through algebraic means.

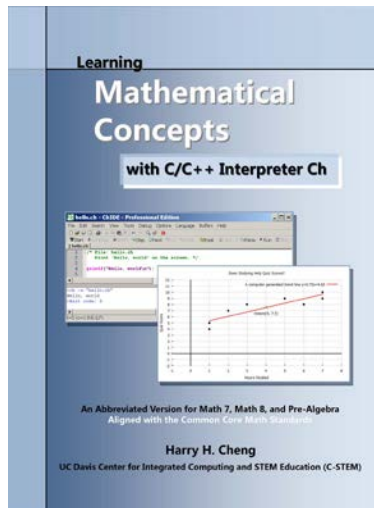
\* *Teaching resources contain optional robotics activities.*



## 5.2 Math 8 with Computing

This course, based on the Common Core Math 8 standards, uses computing to develop and expand students' understanding of Math 8 topics. Students analyze real life situations, identify given information, formulate mathematical steps to find a solution, and analyze the results for accuracy, all within the context of computer programming. The logical process of computer programming allows students to organize their approach to problem solving and efficiently analyze and correct their work. Topics covered include evaluating expressions, one variable equations and inequalities, rates, proportions, probability, scientific notation, statistics, plotting points, linear equations in slope-intercept form, systems of linear equations, radical expressions and equations, similarity, and geometric transformations, including translations and reflections. Optional group computing activities allow students to collaborate on critical thinking activities based on algebraic topics while developing their ability to effectively communicate, listen, share responsibility and respectfully address the suggestions of others. Optional robotics extension activities allow students to reenact physically derived mathematical problems through robotics technologies to visualize situations, associate graphs with physical phenomenon, predict and identify key features of the graphs with the specific physical situations, and solve physical problems' through algebraic means.

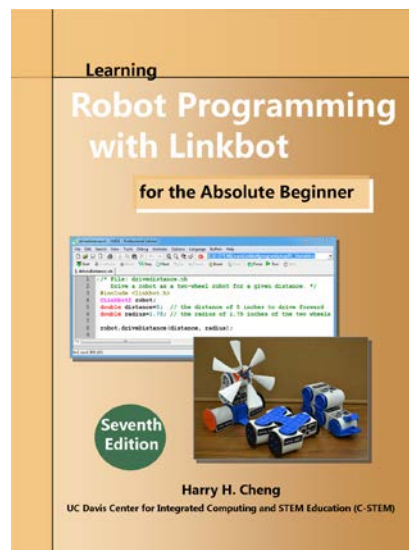
\* Teaching resources contain optional robotics activities.



### 5.3 Robotics and Film Production

This course introduces students to the working principles of robotics with applications for film production using robotics. Students will explore fun applications, such as robotic soccer and robotic drawing by controlling a single robot out of the box, and continue on to multiple robot applications aided by a graphical user interface and computer programming using the C/C++ interpreter Ch. Students write robotics programs to perform various tasks with applications for the RoboPlay competition. With robots, students explore their creativity in writing, art, music, choreography, design, video editing and film production. This course emphasizes hands-on robotics activities to explore applications of robotics to gain effective communication and team work skills.

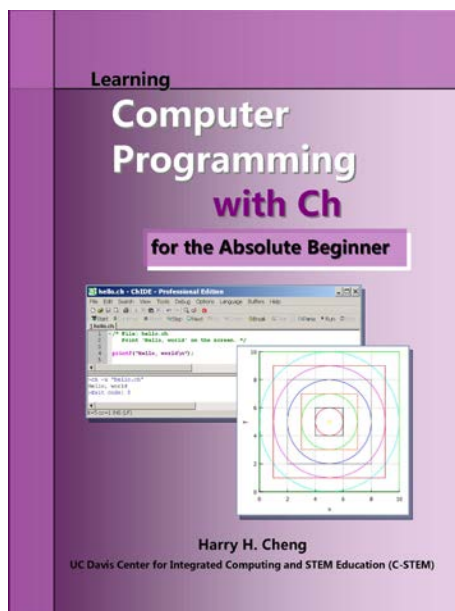
\* This course can be implemented as a standalone robotics course or as a supplement to a Physical Science or Engineering course.



## 5.4 Introduction to Computer Programming

This course introduces students to the fundamentals of computer programming with an emphasis on applications of math concepts using the user friendly C/C++ interpreter Ch. Students start with basics of how a computer works and then explore programming in Ch to solve real life problems. Students write computer programs with graphical plotting and animation in an integrated development environment (IDE). Through computer programming based problem solving and engaging activities, such as generating random numbers for applications in math and gaming, students develop critical and computational thinking skills. Each section includes objectives, pre-requisites, applicable Common Core Language, Reading and CTE ICT standards, terminology, text with examples and applications, and exercises.

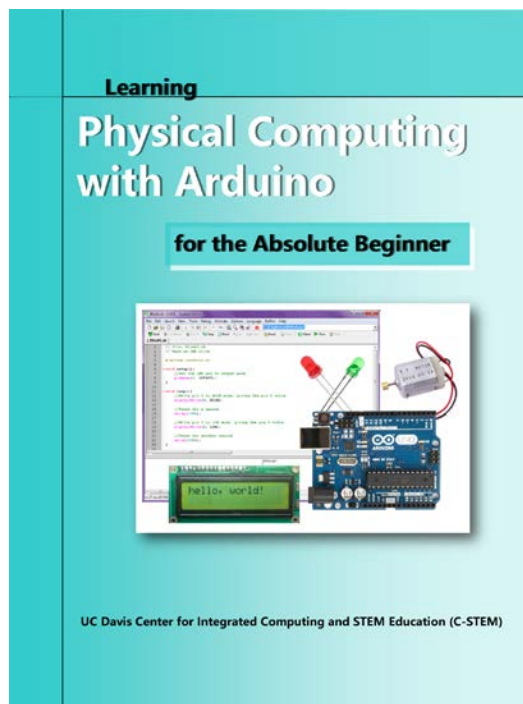
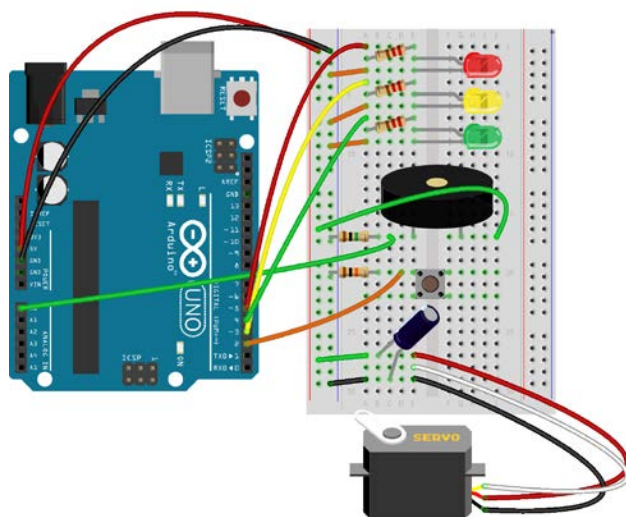
*\* Teaching resources contain optional robotics activities.*



## 5.5 Introduction to Physical Computing and Making

This course introduces students to physical computing and making. Students learn how to program Arduino with a user-friendly graphical user interface (GUI) ChDuino and C/C++ interpreter Ch. Students also learn basics of electronics and how Arduino responds to sensors using enlightening examples. Students learn data acquisition and visualization of experimental data using Arduino for science and engineering projects. Then, students design and build more fun prototypes based on their imagination as team projects. At the end, student teams present their projects.

**\* Approved with G elective credit as a College-Preparatory Elective in Math and Computer Science.**



## 6 Catalog Information for High School A-G Courses for the C-STEM Math-ICT Curriculum

The course outlines and unit information for these A-G approved C-STEM courses by UCOP are available at:

<http://c-stem.ucdavis.edu/curriculum/high-school/>

Teacher credentials to teach these A-G courses are described at:

<http://c-stem.ucdavis.edu/curriculum/high-school/c-stem-a-g-credential-requirements/>

### 6.1 Algebra 1 with Computing (1 year)

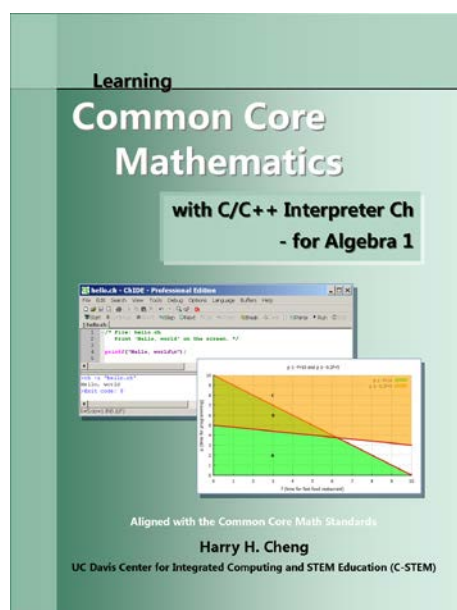
This course guides students through topics in Algebra 1 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch to reinforce and extend their knowledge of mathematical concepts. This is done 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 or modifying the programming solutions as necessary. Topics covered include:

- solving one-variable equations with multiple steps,

- solving and plotting absolute value equations and inequalities,
- linear equations,
- systems of linear equations and inequalities,
- polynomial functions,
- exponential functions,
- step and piecewise functions,
- evaluating, multiplying, and factoring polynomial functions,
- solving quadratic equations with applications,
- probability,
- statistical data analysis and visualization, and
- arithmetic and geometric sequences

Group computing projects allow students to collaborate on critical thinking activities based on algebraic topics while developing their teamwork and communication skills.

\* **Approved with C math credit.** *Teaching resources contain optional robotics activities.*



## 6.2 Algebra 1 with Computing and Robotics (1 year)

The course guides students through topics in Algebra 1 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch to reinforce and extend their knowledge of mathematical concepts. This is done 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. Topics covered include:

- solving one-variable equations with multiple steps
- solving and plotting absolute value equations and inequalities
- linear equations
- systems of linear equations and inequalities
- polynomial functions
- exponential functions
- step and piecewise functions
- evaluating, multiplying, and factoring polynomial functions
- solving quadratic equations with applications
- probability
- statistical data analysis and visualization, and
- arithmetic and geometric sequences

Robotics activities allow students to reenact physically derived mathematical problems through robotics technologies in order to visualize situations, associate linear and quadratic graphs with physical phenomenon, predict and identify key features of graphs with robotic systems, and solve robotics problems through mathematical modeling and programming.

**\* Approved with C math credit.** *Teaching resources contain robotics activities.*

## 6.2 Geometry with Computing and Robotics (1 year)

The course guides students through topics in Geometry in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch 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. Topics covered include applications using area and perimeter, parallel and perpendicular lines, transformations, congruent triangles, quadrilaterals and other polygons, similarity, right triangles and trigonometry, coordinate proofs, circles, circumference, area, volume, and probability. Robotics technology will be used to introduce and expand upon the areas of study listed above. Robotics activities allow students to reenact physically derived mathematical problems to visualize situations, associate graphs with physical phenomenon, apply geometric and trigonometric properties and solve, and solve robotics problems through mathematical modeling and programming.

\* Approved with C math credit. *Teaching resources contain robotics activities.*

### 6.3 Algebra 2 with Computing and Robotics (1 year)

The course guides students through topics in Algebra 2 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch 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. Students focus on applying methods from probability and statistics to draw inferences and conclusions from data, expanding understanding of functions to include square roots, cube roots, absolute values, piecewise, step, polynomial, exponential, rational, radical, logarithmic, and trigonometric functions. Robotics technology will be used to introduce and expand upon the areas of study listed above. Robotics activities allow students to reenact physically derived mathematical problems 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.

\* Approved with C math credit. *Teaching resources contain robotics activities.*

### 6.4 Integrated Mathematics 1 with Computing (1 year)

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. Students use programming in C/C++ interpreter Ch to reinforce and extend their knowledge of mathematical concepts. This is done 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. Topics covered include:

- solving one-variable equations with multiple steps
- solving and plotting absolute value equations and inequalities
- linear equations
- systems of linear equations and inequalities
- exponential functions
- statistical data analysis and visualization

- arithmetic and geometric sequences
- geometric transformations, including translations, rotations, and reflections
- geometric construction

Group computing projects allow students to collaborate on critical thinking activities based on mathematics topics while developing their teamwork and communication skills.

\* **Approved with C math credit.** *Teaching resources contain optional robotics activities.*

### **6.5 Integrated Mathematics 1 with Computing and Robotics (1 year)**

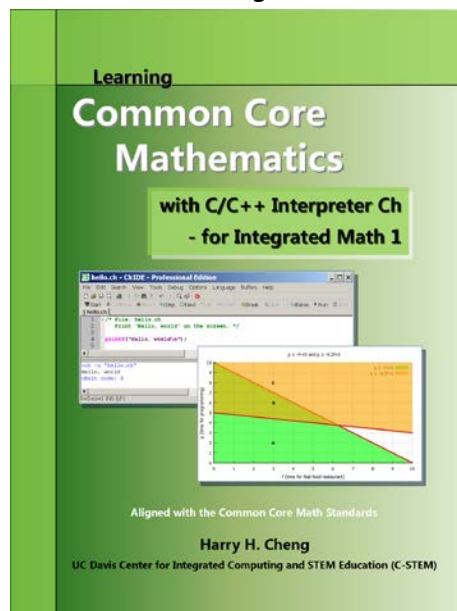
The course guides students through topics in Integrated Mathematics 1 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch to reinforce and extend their knowledge of mathematical concepts. This is done 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. Topics covered include:

- solving one-variable equations with multiple steps
- solving and plotting absolute value equations and inequalities
- linear equations
- systems of linear equations and inequalities
- exponential functions
- statistical data analysis and visualization
- arithmetic and geometric sequences
- geometric transformations, including translations, rotations, and reflections
- geometric construction

Robotics activities allow students to reenact physically derived mathematical problems through robotics technologies in order to visualize situations, associate linear and exponential graphs with physical phenomenon, predict and identify key features of graphs with robotic systems, and solve robotics problems through mathematical modeling and programming.



\* **Approved with C math credit.** *Teaching resources contain robotics activities.*



## 6.6 Integrated Mathematics 2 with Computing and Robotics (1 year)

The course guides students through topics in Integrated Mathematics 2 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch 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. Topics covered include recognizing and developing patterns using tables, graphs and equations. Mathematical modeling is stressed as a methodology for approaching the solution to problems. Students will explore operations on algebraic expressions, and apply mathematical properties to algebraic equations. Students will problem solve using equations, graphs and tables and investigate linear relationships, including comparing and contrasting options and decision-making using algebraic models. Reinforcement of topics from two-dimensional geometry is integrated into this curriculum. This includes applications from the areas and perimeters, the Pythagorean Theorem and its applications, as well as geometric proportion. Finally, introductory instruction in the area of mathematical probability is provided to reinforce numerical modeling. Robotics technology will be used to introduce and expand upon the areas of study listed above. Robotics activities allow students to reenact physically derived mathematical problems 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.

\* **Approved with C math credit.** *Teaching resources contain robotics activities.*

### 6.7 Integrated Mathematics 3 with Computing and Robotics (1 year)

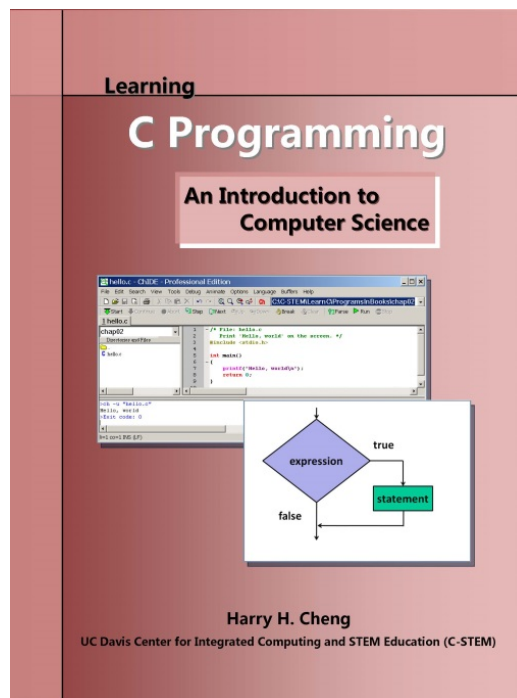
The course guides students through topics in Integrated Mathematics 3 in Common Core State Standards for Mathematics while simultaneously teaching students programming and computational thinking. Students use programming in C/C++ interpreter Ch to reinforce and extend their knowledge of mathematical 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. Students focus on applying methods from probability and statistics to draw inferences and conclusions from data, expanding understanding of functions to include square roots, cube roots, absolute values, piecewise, step, polynomial, exponential, rational, radical, logarithmic, and trigonometric functions, expanding right triangle trigonometry to include general triangles, and consolidating functions and geometry to create models and solve contextual problems. Robotics technology will be used to introduce and expand upon the areas of study listed above. Robotics activities allow students to reenact physically derived mathematical problems 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.

\* Approved with C math credit. ***Teaching resources contain robotics activities.***

## 6.8 Computer Programming for Solving Applied Problems (1 year)

This course provides students with the fundamental knowledge of computer programming for solving applied problems in C. Students learn how a computer works and structured programming in C for software development. The topics include programming constructs, data types and declaration of variables, expressions and operators, selection statements, repetition, flowcharts for algorithm development, functions for modular programming, arrays for statistical data analysis, plotting for visualizing data (using scatter plot, dot plot, bar graph, histogram, Box-and-Whisker plot, etc.), linear regression and curve fitting, processing data files, animation, robotics applications, and applications in math and science. The emphasis of the course is to introduce the students to software development concepts. This course also focuses on algorithm development and computer programming for solving applied problems in science, technology, engineering and math (STEM), such as solving problems in Algebra and robotics. Considerable attention is devoted to program design, task decomposition, testing, debugging, and software reuse. Students write computer programs with graphical plotting in an integrated development environment. Through problem-based projects, students develop critical thinking, problem solving, computational thinking, effective communication, and teamwork skills.

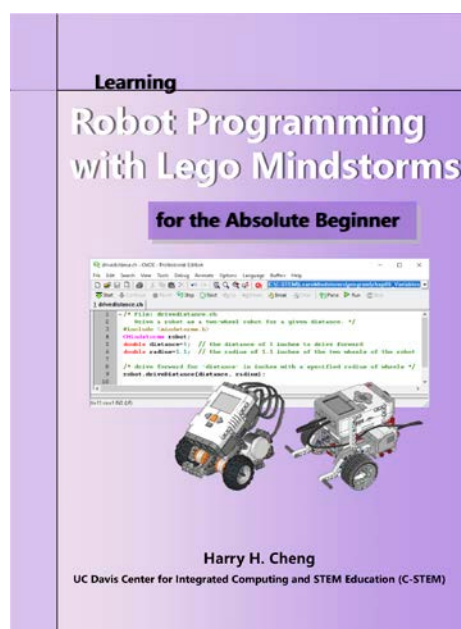
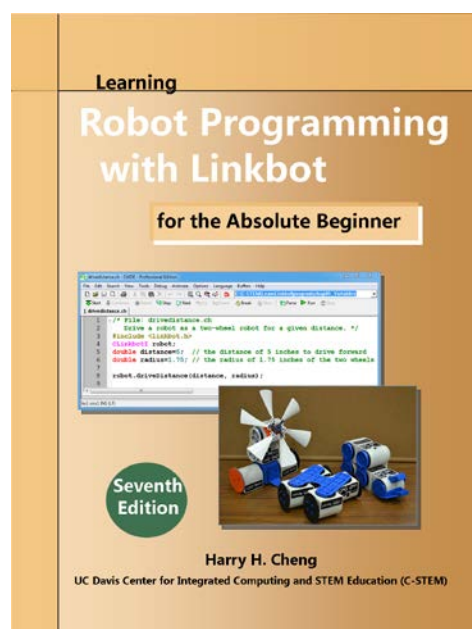
**\* Approved with G elective credit as a College-Preparatory Elective in Math and Computer Science.** *Teaching resources contain robotics activities.*



## 6.9 Computing with Robotics (1 year)

This course introduces students to the working principles and foundational knowledge of robotics. Students learn to control a single robot and multiple robots by graphical user interface and computer programs in C/C++. Students write robotics programs to perform various tasks based on the sensory information of robots. Robots are used as platforms to engage students in both personalized and collaborative learning computing, science, technology, engineering, and math concepts. This course emphasizes hands-on robotics activities with a concentration on mathematical modeling and computer programming for solving problems in math and science. As team projects, students will participate in regional and statewide C-STEM RoboPlay Video and/or RoboPlay Challenge Competitions, which not only enhance their learning of robotics, math, and engineering, but also allow them to explore their creativity in writing, art, music, choreography, design, video editing, and film production. Through these project-based team activities, students develop critical thinking, problem solving, effective communication, and teamwork skills.

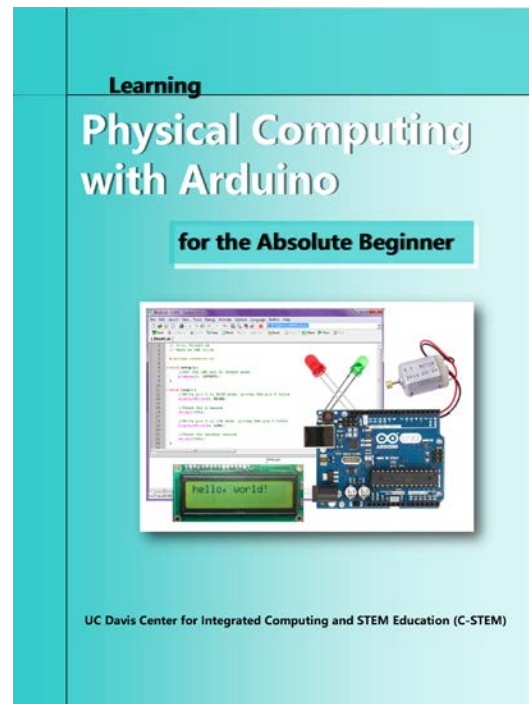
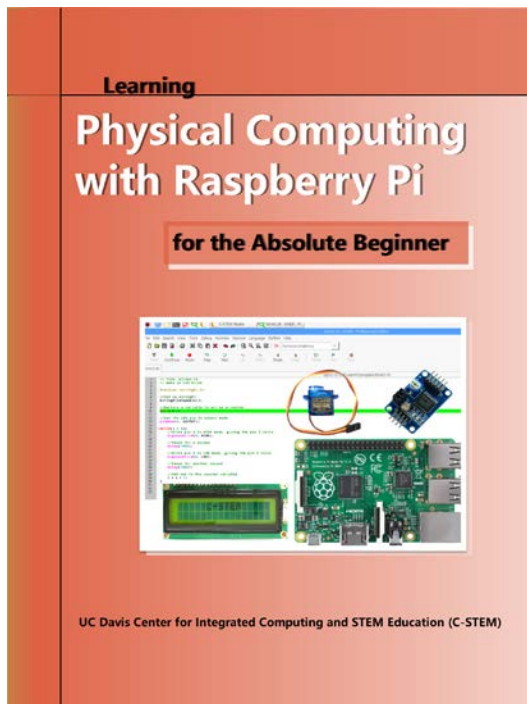
**\* Approved with G elective credit as a College-Preparatory Elective in Math and Computer Science. (We will submit it for approval with math "C" credit)**



## 6.10 Physical Computing with Pi and Arduino (one semester)

This one-semester course provides students with the fundamental knowledge of physical computing and making. Students learn how a computer works and structured programming in C for software development for interfacing electronics and sensors using ultra-low-cost computers of Arduino and Raspberry Pi (Pi). The topics include programming constructs, data types and declaration of variables, expressions and operators, selection statements, repetition, functions for modular programming, arrays for statistical data analysis, plotting for visualizing data, processing data files, real-world application projects. The emphasis of the course is to introduce the students to software development concepts for interface with hardware using Arduino and Pi. Students learn how to program Arduino and Pi with a user-friendly graphical user interface (GUI) Chduino and GPIOviewer, and C/C++ interpreter Ch, as well as Arduino IDE. Students also learn basics of electronics and how Arduino responds to sensors using enlightening examples. Then, students design and build more fun prototypes based on their imagination as team projects. At the end, student teams present their projects.

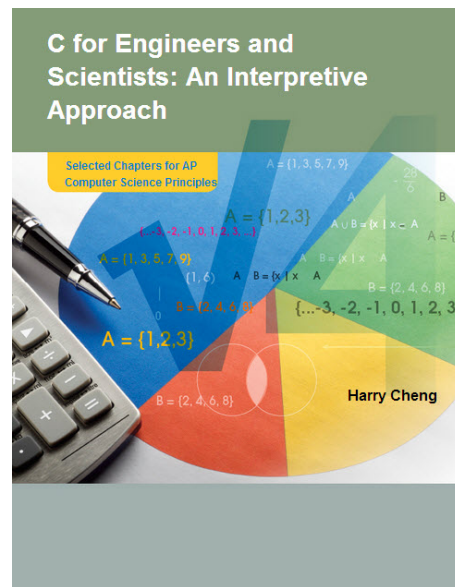
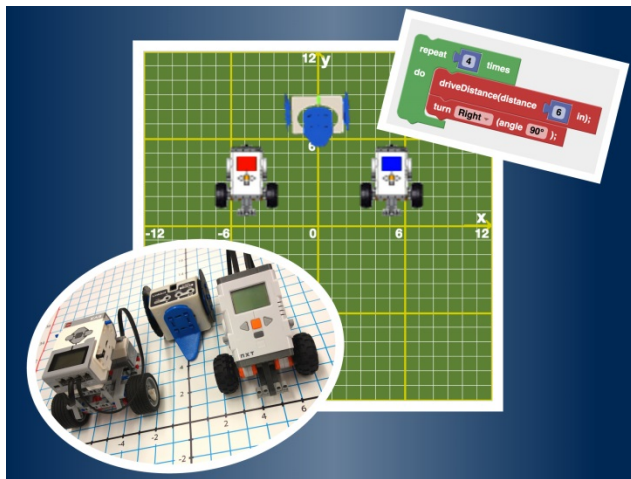
**\* Approved with G elective credit as a College-Preparatory Elective in Math and Computer Science.**



## 6.11 AP Computer Science Principles (Pilot teaching) (1 year)

Using the C/C++ interpreter Ch as a platform for computation, this course introduces students to computer science principles aligned with the learning objectives described in the College Board's CS Principles Curriculum Framework and prepare students to take the 2017 AP CS Principles exam. Students learn computational thinking and software development for practical applications using both basic knowledge of number systems, data types, functions, selection and iteration statements and advanced programming concepts such as, pointers for dynamic memory allocation, structures, and linked lists in C. Students explore the working principles of the internet, visualization of data, cyber-security, simulation, and film production. Students will also learn the intelligent behavior of robotic systems through software implementation.

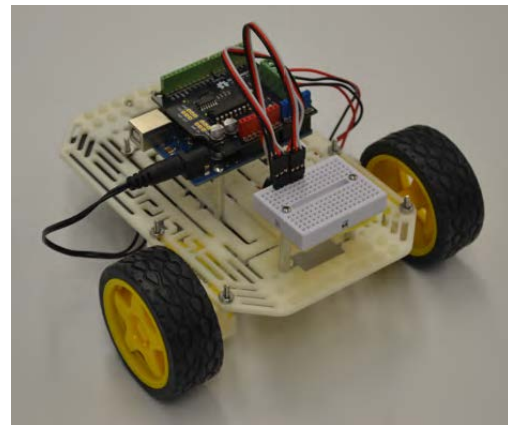
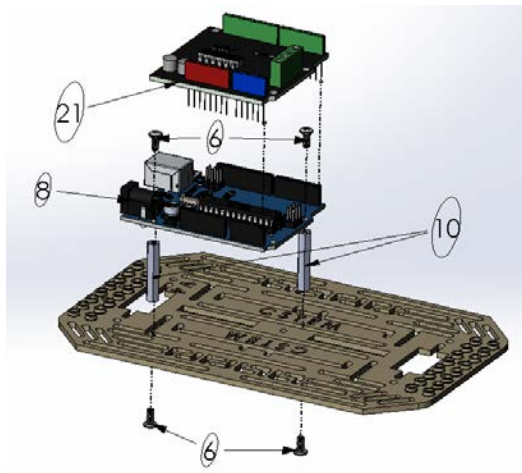
**To be approved with G elective credit as a College-Preparatory Elective in Math and Computer Science.**



## 6.12 Principles and Design of Cyber-Physical Systems (Pilot teaching) (1 year)

Cyber-physical systems (CPS) are engineered systems that are built from, and depend upon, the seamless integration of computational algorithms and physical components. This course gives students an introduction to Cyber-Physical Systems using CPSkit. CPSkit is a versatile robot kit, specially designed to teach principles and design of cyber-physical systems. Students can build a two-wheel robot and other cyber-physical systems with CPSkit using off-the-shelf components and 3D printed parts. The system is controlled by a Raspberry Pi microcomputer and an Arduino microcontroller with a breadboard for connecting electronic parts without soldering. All mechanical components of the CPSkit can be 3D printed. The system can also be connected with other robotics components such as Linkbot and Lego Mindstorms NXT or EV3 robots for various applications. CPSkit can be controlled conveniently through a user-friendly C/C++ interpreter Ch. Students learn how to program Arduino and Pi with a user-friendly graphical user interface (GUI) ChDuino and GPIOviewer, and C/C++ interpreter Ch, as well as Arduino IDE. Students learn design thinking by using the CPSkit to build their own CPS. The hands-on CPS curriculum will excite students' imagination and foster their interest in computing and cyber-physical systems, as well as allow them to join the Maker Movement. The course includes the following eight units: Computer Programming in Raspberry Pi in RoboBlockly and Ch/C/C++, Electronics in Arduino and Raspberry Pi, Mathematical Modeling, Engineering Design, Programming CPSbot, Communication, optional 3D Modeling and 3D Printing, and RoboPlay Video Competition with CPS.

**\*Approved with G elective credit as a College-Preparatory Elective in Math and Computer Science.**



A robot built using a CPSkit.

