

Coding in Math Syllabus

Independent, Interdisciplinary Math Modules

Course Overview and Goals

Students don't need to be in a computer science class to explore coding. There are coding applications in every subject, including art, math, science and history.

With each module in this course, students use coding to reinforce and extend their understanding of **mathematics!** As they learn major programming concepts, they will develop math-related projects that demonstrate their proficiency in both computer science and math. These modules are a great way to introduce students to programming and show them that coding is a valuable skill in a variety of disciplines.

Learning Environment

Each module in this course is a standalone, web-based curriculum. They contain a series of lessons that introduce students to coding and extend or reinforce a particular math concept. The lessons include short video tutorials, example programs, quizzes, programming exercises, and challenge problems.

Programming Environment

Students write and run programs in the browser using the CodeHS online editor.

Assessments

Each lesson includes at least one formative short multiple choice quiz that helps reinforce major concepts. At the end of each module, students will complete a final project and/or take a summative multiple choice quiz that tests their knowledge of the concepts covered in the module.

Prerequisites

These interdisciplinary modules are designed for students with no previous background in computer science. Each module varies in topic and difficulty depending on recommended grade level. The modules are highly visual, dynamic, and interactive, making them an engaging introduction to computer science.

More Information

Browse the content of this course at <https://codehs.com/course/13419>

Module Breakdown

The modules below are all independent of one another -- they can be used on their own, in any order. They span a variety of topics and grade levels.

Module Title	Description	Major Topics	Grade	Timing
Sweet Factors	Students will leverage computational power to find all of the factors of a number in order to help The Sweet Shop bag its homemade candies.	Math: factors, greatest common factor, (multiples, least common multiple) CS: printing, modulus arithmetic, variables, if statements, for loops	MS	5-8 hours (an additional 3-4 hours for extension content)
Shapes and Pythagorean Theorem	Students will learn to code with variables and user input in order to develop a final program that determines the dimensions of a roof and the needed construction materials.	Math: area and perimeter, pythagorean theorem CS: printing, arithmetic, variables, user input	MS	4-6 hours
The Formula Solver	Students learn to build programs that solve various real-life formulas based on a user's input.	Math: algebraic manipulation, variables, real-life problems/formulas CS: printing, arithmetic, variables, user input, if/else statements, comparative operators	HS	3-5 hours
Vector Math	Students reinforce their understanding of vector measurements as they learn to program with them in a variety of physical applications.	Math: vectors, components, trigonometry, right triangles CS: printing, arithmetic, variables, arrays	HS	3-5 hours
Categorizing Triangles	Students will use Turtle Graphics in Python to create a program that draws and categorizes triangles based on user input.	Math: triangles, area, algebra CS: variables, arithmetic, if statements, comparative and logical operators, functions	MS	1-3 hours
Building Mathematical Models	Students use Tracy the Turtle to learn how to code different mathematical models in Python! Note: Students should have completed Algebra I or higher.	Math: x-y coordinates, equation plotting, mathematical modeling CS: variables, arithmetic, functions, for loops	HS	1 hour
Orbital Precision	Students explore the impact of using more digits of Pi when calculating the orbit of Nasa	Math: pi calculations, precision, percent error, unit conversions	HS	1-2 hours

	Satellites. Note: Students should be familiar with basic loops, conditional, and print statements in Java before attempting.	CS: printing, variables, arithmetic, methods, for loops, if/else statements, logical and comparison operators.		
--	--	---	--	--