



CodeHS

Indiana Computer Science I Course Syllabus 1 year for High School (145-155 contact hours)

Course Overview and Goals

The CodeHS Computer Science I curriculum teaches the foundations of computer science and basic programming, with an emphasis on helping students develop logical thinking and problem solving skills. Once students complete the CodeHS Computer Science I course, they will have learned material equivalent to a semester college introductory course in Computer Science and be able to program in JavaScript.

Learning Environment: The course utilizes a blended classroom approach. The content is fully web-based, with students writing and running code in the browser. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students. Each unit of the course is broken down into lessons. Lessons consist of video tutorials, short quizzes, example programs to explore, and written programming exercises, adding up to over 100 hours of hands-on programming practice in total. Each unit ends with a comprehensive unit test that assesses student's mastery of the material from that unit as well as challenge problems where students can display their understanding of the material.

Programming Environment: Students write and run JavaScript programs in the browser using the CodeHS editor.

More information: Browse the content of this course at <https://codehs.com/course/5639>

Prerequisites: The Computer Science I course is designed for complete beginners with no previous background in computer science. The course is highly visual, dynamic, and interactive, making it engaging for new coders.

Course Breakdown

Unit 1: Introduction to Programming in JavaScript with Karel the Dog (3 weeks/15 hours)

Module one is programming with Karel. Karel is a dog that only knows how to move, turn left, and place tennis balls in his world. You can give Karel these commands to instruct him to do certain things. We use Karel to show you what it means to program, and allow you to focus on problem solving.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7772>

Objectives / Topics Covered	<ul style="list-style-type: none">● Commands● Defining vs. Calling Methods● Designing methods● Program entry points● Control flow● Looping● Conditionals● Classes● Commenting code● Preconditions and Postconditions● Top Down Design
-----------------------------	---

Assignments / Labs	<ul style="list-style-type: none"> ● 26 Karel Programming Exercises and Challenges in total ● Program-specific tasks for Karel the Dog <ul style="list-style-type: none"> ○ Example Exercise: Pyramid of Karel Write a program to have Karel build a pyramid. There should be three balls on the first row, two in the second row, and one in the third row. ● Teach Karel new commands like <code>turnRight()</code> or <code>makePancakes()</code> <ul style="list-style-type: none"> ○ Example Exercise: Pancakes Karel is the waiter. He needs to deliver a stack of pancakes to the guests on the 2nd, 4th, and 6th avenue. Each stack of pancakes should have three pancakes. Create a method called <code>makePancakes()</code> to help Karel solve this problem. ● Solve large Karel problems by breaking them down into smaller, more manageable problems using Top Down Design <ul style="list-style-type: none"> ○ Example Exercise: The Two Towers In this program, Karel should build two towers of tennis balls. Each tower should be 3 tennis balls high. At the end, Karel should end up on top of the second tower, facing East. ● Using control structures and conditionals to solve general problems <ul style="list-style-type: none"> ○ Example Exercise: Random Hurdles Write a program that has Karel run to the other side of first street, jumping over all of the hurdles. However, the hurdles can be in random locations. The world is fourteen avenues long.
--------------------	--

Unit 2: Karel Challenges (1.5 weeks, 7 hours)

In this module you'll take all the foundational concepts from Karel to solve some programming challenges.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7773>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Solving large and more complex problems using Karel
Assignments / Labs	<ul style="list-style-type: none"> ● 5 Karel challenges to tie everything learned in the Karel module together <ul style="list-style-type: none"> ○ Example Exercise: Super Cleanup Karel Karel's world is a complete mess. There are tennis balls all over the place, and you need to clean them up. Karel will start in the bottom left corner of the world facing east, and should clean up all of the tennis balls in the world. This program should be general enough to work on any size world with tennis balls in any locations.

Unit 3: Cybersecurity (7 weeks, 35 hours)

This module includes topics on Internet etiquette and how to stay safe on the world wide web. We will also look at the potential effects of our digital footprints, how to protect information from online risks, and the implications of hacking. Finally, the module includes how to find and cite quality resources online.

Browse the full content of this module at <https://codehs.com/library/course/5639/module/7798>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Digital Footprint and Reputation ● Privacy and Security ● Information Literacy ● Creative Credit and Copyright ● Hacking Ethics
Example Assignments	<ul style="list-style-type: none"> ● Digital Footprint and Reputation

/ Labs	<ul style="list-style-type: none"> ○ Example activities: <ul style="list-style-type: none"> ■ What is your digital footprint? ■ Are you going to make any changes in what you post on social media? ● Keeping data private and secure <ul style="list-style-type: none"> ○ Example activities: <ul style="list-style-type: none"> ■ Test out various passwords on a site ■ Explore Google’s privacy policy: What do they know about you? ● Information Literacy <ul style="list-style-type: none"> ○ Example activities: <ul style="list-style-type: none"> ■ Create and test search queries ■ Explore evidence for using sources ● Different types of copyright licenses <ul style="list-style-type: none"> ○ Example activities: <ul style="list-style-type: none"> ■ Create citations for sources ■ Explore image search tools ● Hacking Ethics <ul style="list-style-type: none"> ○ Example activities: <ul style="list-style-type: none"> ■ Explore what penetration testing is ■ Sign ethical hacker agreement
--------	--

Unit 4: Javascript & Graphics (1.5 weeks/7 hours)

Introduces you to the basics of JavaScript, including variables, user input, mathematics, basic graphics, and image representations.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7798>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Variables ● User Input ● Arithmetic Expressions ● Graphics ● Image Representation
Assignments / Labs	<ul style="list-style-type: none"> ● 7 JavaScript & graphics programming exercises in total ● Using variables and getting user input using JavaScript <ul style="list-style-type: none"> ○ Example Exercise: Dinner Plans Prompt the user for their name, then ask them what time you should meet for dinner. Greet them by name and tell them you will meet them at the time they specified! ● Create basic graphics in JavaScript <ul style="list-style-type: none"> ○ Example Exercise: Flag of the Netherlands This program should draw the flag for the Netherlands. The top third of the canvas is red, the middle third is white, and the bottom third is blue, as shown below. You will need to use Rectangle objects in this program.

Unit 5: Graphics Challenges (1 week, 5 hours)

In this module you'll learn how to pair program! You'll also take all the foundational concepts from JavaScript Graphics to solve some programming challenges.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7801>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Pair programming ● Solving large and more complex problems that incorporate graphics and user input
Assignments / Labs	<ul style="list-style-type: none"> ● 3 graphics challenges to tie everything learned in the JavaScript & Graphics module together <ul style="list-style-type: none"> ○ Example Exercise: Ghost Write a program to draw a ghost on the screen. You must do this by using the constant values given (this will allow us to easily alter the size or color of the ghost.)

Unit 6: JavaScript Control Structures (3 weeks/15 hours)

Learn how to use control structures such as if/else statements and loops to make more advanced programs in JavaScript.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7775>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Booleans ● For Loops ● Conditionals ● Nested Control Structures ● While Loops
Assignments / Labs	<ul style="list-style-type: none"> ● 22 control structures programming exercises in total ● Using comparison and logical operators to control the flow of the program <ul style="list-style-type: none"> ○ Example Exercise: Inventory Write a program that keeps track of a simple inventory for a store. While there are still items left in the inventory, ask the user how many items they would like to buy. Then print out how many are left in inventory after the purchase. You should use a while loop for this problem. Make sure you catch the case where the user tries to buy more items than there are in the inventory. In that case, you should print a message to the user saying that their request isn't possible. ● Using for loops <ul style="list-style-type: none"> ○ Example Exercise: All Dice Values Write a program that prints all possible dice rolls with 2 dice To do so, you should use a double for loop. ● Drawing basic graphics using control structures <ul style="list-style-type: none"> ○ Example Exercise: Caterpillar This graphics program should draw a caterpillar. A caterpillar has NUM_CIRCLES circles. Every other circle is a different color, the even circles are red, and the odd circles are green (by even we mean when i is an even number). Use a for loop to draw the caterpillar, centered vertically in the screen. Also, be sure that the caterpillar is still drawn across the whole canvas even if the value of NUM_CIRCLES is changed.

Unit 7: Control Structures Challenges (1 week, 5 hours)

In this module you'll take all the foundational concepts from the Control Structures unit to solve some programming challenges.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7802>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Solving large and more complex problems that incorporate control structures
-----------------------------	---

Assignments / Labs	<ul style="list-style-type: none"> ● 4 challenges using control structures to tie everything learned in the JavaScript Control Structures module together <ul style="list-style-type: none"> ○ Example Exercise: Circles in Squares Repeatedly draw a circle inside of a square and then a square inside that circle and so on. You should repeat this while the size is greater than MIN_SIZE. Each time you draw a shape, it should be centered on the screen and a new random color. You'll need to update the size based on the size of the previous one.
--------------------	--

Unit 8: Functions and Parameters (2 weeks, 10 hours)

Learn to write reusable code with functions and parameters.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7776>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Functions with and without parameters ● Functions with and without return values ● Nested Control Structures ● Local variables and scope
Assignments / Labs	<ul style="list-style-type: none"> ● 14 functions programming exercises in total <ul style="list-style-type: none"> ○ Example Exercise: Vertical Lines Write a function that draws vertical lines on the graphics canvas. If a line is vertical, then the x-values for the endpoints are the same. The parameters to your function should be the x location, and the length, and all of your lines should start at y position 0. ○ Example Exercise: Is it even? Write a function called isEven that returns a boolean of whether or not a value is even or odd. The isEven function should not print anything out or return a number. It should only take in a number and return a boolean. Once you've written this function, write a program that asks the user for integers and prints whether the number they entered is even or odd using your isEven function. You should let the user keep entering numbers until they enter the SENTINEL given.

Unit 9: Functions Challenges (1 week, 5 hours)

Use what you have learned in the Functions unit to solve challenges.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7777>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Solving large and more complex problems that incorporate functions, parameters, and return values
Assignments / Labs	<ul style="list-style-type: none"> ● 4 challenges using functions to tie everything learned in the Functions and Parameters module together <ul style="list-style-type: none"> ○ Example Exercise: Grades Write a function that takes a number score from 0 to 100 and prints the grade. You should print the letter grade A-F and the +/- sign if needed

Unit 10: Animation and Games (3 weeks, 15 hours)

Now, your graphics will come to life. You will learn how to make objects move around the screen. You will also learn how to let the user interact with your program with the mouse. At the end of this section, you will program your very own video game.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7778>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Timers ● Randomizing Games ● Mouse Events ● Keyboard Events
Assignments / Labs	<ul style="list-style-type: none"> ● 15 animations programming exercises in total ● Using timers to add randomizations to graphical programs <ul style="list-style-type: none"> ○ Example Exercise: Paint Splatter Write a program that splatters paint on the screen every DELAY milliseconds. To splatter paint, pick a random color and draw CIRCLES_PER_SPLATTER circles of that color at random places on the screen. The radius of each circle should be a random value between MIN_RADIUS and MAX_RADIUS. Remember to use helper functions. ● Using mouse events for interactive programs <ul style="list-style-type: none"> ○ Example Exercise: Teleporting Ball Extend our bouncing ball program. Whenever you click, the ball should teleport to that spot and change to a random color. ○ Example Exercise: Target Draw a target on the screen that moves to aim at where your mouse is located. A target consists of a horizontal line that goes from 0 to the window width and a vertical line that goes from 0 to the window height. The lines should cross paths where the mouse is. ● Using keyboard events for interactive programs <ul style="list-style-type: none"> ○ Example Exercise: Basic Snake Write a basic version of the snake game. The way our game works is by first creating a green square at the center of the screen. The snake should be moving to the right. If you hit an arrow key, you should change the snake's direction.

Unit 11: Animations Challenges (1 week, 5 hours)

In this module you'll take all the foundational concepts from the Animations unit to solve some programming challenges.

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7806>

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Solving large and more complex problems that incorporate animation and timers
Assignments / Labs	<ul style="list-style-type: none"> ● 2 challenges using animation to tie everything learned in the Animation & Games module together <ul style="list-style-type: none"> ○ Example Exercise: Blinking Rectangles You should divide the canvas into an imaginary grid with NUM_RECTANGLES_ACROSS rectangles across, and NUM_RECTANGLES_DOWN rectangles down. Each time the user moves the mouse, a rectangle aligned with this grid should be drawn so that the mouse's location is within the rectangle. The rectangle should change color each time the mouse passes over it. This requires using the <code>mouseMoveMethod</code> as well as writing a function.

Unit 12: Project: Breakout (2 weeks, 10 hours)

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7779>

Objectives / Topics Covered	<ul style="list-style-type: none">● Basic graphics● Mouse events● Collision detection
Assignments / Labs	<ul style="list-style-type: none">● Guided exercises to build a Breakout Game● The Breakout Game is made up of bricks at the top of the screen, a paddle that you control at the bottom of the screen, and a ball that bounces around. Your goal is to direct the paddle with your mouse to bounce the ball until all of the bricks have been hit and disappear.

Unit 13: Final Project (2-4 weeks, 10-20 hours)

In this module you'll take all the skills you've learned throughout the course to create a final program with a partner!

Browse the full content of this unit at <https://codehs.com/library/course/5639/module/7807>

Objectives / Topics Covered	<ul style="list-style-type: none">● Project Planning● Final Project Creation
Assignments / Labs	<ul style="list-style-type: none">● Collaborative open-ended final project which encourages creativity● Program Requirements: Your program:<ul style="list-style-type: none">○ must use JavaScript Graphics○ must allow the user to interact with your project with either their mouse or keyboard○ must use at least one timer○ must break down the program into multiple functions○ must utilize control structures where applicable

Optional Supplemental Materials (Remainder of school year)

Objectives / Topics Covered	<ul style="list-style-type: none">● Extra practice with:<ul style="list-style-type: none">○ Karel○ Basic JavaScript○ JavaScript functions○ Graphics○ Animation● Basic Data Structures in JavaScript● Game Design● Music Visualization
Assignments / Labs	<ul style="list-style-type: none">● Several additional exercises and large projects covering the topics listed above