CodeHS

Arkansas Programming I
Course Syllabus

Course Overview and Goals

This is the level one computer science course for the state of Arkansas. It is designed to provide foundational understandings of concepts in computer science that are necessary for students to function in an ever-changing technological world. These standards help students learn to accomplish tasks and solve problems independently and collaboratively. These standards give students the tools and skills needed to be successful in college and careers, whether in computer science or in other fields.

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.

Programming Environment: Students write and run JavaScript programs in the browser using the CodeHS editor. Classes can choose to write code using either blocks or text.

More information: Browse the content of this course at https://codehs.com/course/16725

Prerequisites

The Arkansas Programming 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: Digital Citizenship and Cyber Hygiene (5 weeks/25 hours)
This unit focuses on why cybersecurity is important, recent threats to cybersecurity, and different careers in the field. It includes topics on Internet etiquette and how to stay safe on the world wide web. Students will also look at the potential effects of their digital footprints, how to protect information from online risks, and the implications of cyberbullying. Finally, the module includes how to find and cite quality resources online.

Browse the full content of this module at https://codehs.com/course/16725/explore/module/23033

Objectives / Topics	● What is Cybersecurity?

Covered	- The CIA Triad - Digital Footprint and Reputation - Cyberbullying - Internet Safety - Privacy and Security - Information Literacy - Visualizing and Interpreting Data - Data Collection and Limitations - Creative Credit and Copyright - Hacking Ethics - Project: Public Service Announcement
Example Assignments / Labs	- What is Cybersecurity? - Cybersecurity defined - Why is cybersecurity important? - Cybersecurity in the news - Cybersecurity and IoT (Internet of Things) - How do we prevent cyber attacks? - Example activities: - Summarize and discuss recent cyber attacks - Explore a threat map to see where cyber attacks are coming from and which countries are being targeted - The CIA Triad - What is the CIA triad? (confidentiality, integrity, availability) - What are "secure systems?" - What do confidentiality, integrity, and availability mean in cybersecurity? - Example activities: - Determine where scenarios break part of the CIA Triad - Digital Footprint and Reputation - What is a digital footprint? - What is *your* digital footprint and reputation? - What does it mean that the internet is public and permanent? - Who looks at your digital footprint and reputation? - What are some recommended social media guidelines? - How can you maintain your digital footprint? - What does your digital footprint say about you? - Example activities: - What is your digital footprint? - Are you going to make any changes in what you post on social media? - Cyberbullying - What is cyberbullying? - What are the impacts of cyberbullying? - Are there cyberbullying roles? - What do you do if you are being bullied? - What do you do if you see bullying? - How can you be an upstander?

- Example activities:
  - Explore cyberbullying scenarios: What would you do?
- Internet Safety
  - What are some ways to stay safe online?
  - What are some online safety guidelines?
  - Example activities:
    - Explore Internet safety scenarios: What would you do?
- Privacy and Security
  - What are data privacy and security?
  - How can you keep personal data secure and private?
  - What can happen if your data is stolen and what can you do about it?
  - Example activities:
    - Test out various passwords on a site
    - Explore Google's privacy policy: What do they know about you?
- Information Literacy
  - What is information literacy?
  - How can you do effective internet searches?
  - What are some techniques for judging source legitimacy and identifying misinformation?
  - Example activities:
    - Create and test search queries
    - Explore the evidence for using sources
- Creative Credit and Copyright
  - What is copyright?
  - What are the different types of copyright licenses
  - Example activities:
    - Create citations for sources
    - Explore image search tools
- Hacking Ethics
  - What are hackers?
  - Are there different kinds of hackers? (white, black, gray)
  - What are bug bounty programs?
  - Is hacking always illegal?
  - What are the consequences of illegal hacking?
  - Example activities:
    - Explore what penetration testing is
    - Sign an ethical hacker agreement
- Final project: Create a Public Service Announcement
  - Create a Public Service Announcement (PSA) to teach your peers about your selected topic in digital citizenship and cyber hygiene. You can select any of the topics covered in this module. Be creative and make it fun! You could make a video, song, poster, or slideshow.

Unit 2: Programming with Karel (7 weeks, 35 hours)

Karel is a dog that only knows how to move, turn left, and place tennis balls in his world. Students can give Karel these commands to instruct him to do certain things. Karel is used to show students what it means to program, and allows them to focus on problem solving.

Browse the full content of this unit at https://codehs.com/course/16725/explore/module/23034

Objectives / Topics Covered	- Commands - Defining vs. Calling Methods - Designing methods - Abstraction - Program entry points - Control flow - Looping - Conditionals - Classes - Commenting code - Preconditions and Postconditions - Top Down Design
Assignments / Labs	30 Karel Programming Exercises and Challenges in total - Program-specific tasks for Karel the Dog - 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 `turnRight()` or `makePancakes()` - 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 `makePancakes()` to help Karel solve this problem. - Solve large Karel problems by breaking them down into smaller, more manageable problems using Top Down Design - 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 - 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 3: Karel Challenges - optional (2 weeks, 10 hours)
In this unit, students take all the foundational concepts from Karel to solve some programming challenges.

Browse the full content of this unit at

Objectives / Topics Covered	● Solving large and more complex problems using Karel

Assignments / Labs	● Several Karel challenges to tie everything learned in the Karel module together
	     ○ 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 location.

Unit 4: Networks and the Internet (3 weeks/15 hours)
This unit explores the structure and design of the internet, and how this design affects the reliability of network communication, the security of data, and personal privacy.

Browse the full content of this unit at

Objectives / Topics Covered	● Structure of the internet ● How network data is transmitted ● Hardware involved in the transmission of data ● How the internet has impacted everyday life ● Using the Command Line Interface (CLI)

Example Assignments / Labs	● Structure of the internet
	   ○ Explore the differences between IPv4 and IPv6. Why are we running out of addresses?
	   ○ Explore the different levels of the internet.
	   ○ Example Activity
	     ■ Trace a website request from the server, through the network, and to your computer
	 ● How data is transmitted
	   ○ How are internet packets able to find their way to your computer?
	   ○ Explain in your own words how a request from your computer travels through the various levels of servers to reach and return the correct webpage and resources?
	   ○ Example Activity:
	     ■ As a class, create a protocol that will allow one classmate to send another classmate a note, without the need for talking to each other.
	 ● Hardware involved
	   ○ Explore the role of routers
	   ○ Why are protocols so important?
	   ○ Explore how data is able to be transmitted across the ocean by using underwater cables

Module 5: Project: Troubleshooting Project (1 week/5 hours)
Students will explore the troubleshooting methodology and utilize it to solve sample IT support issues.

Browse the full content of this module at

Objectives / Topics Covered	- Troubleshooting Methodology - Identify the problem - Research past solutions - Establish a theory - Test the theory - Establish a plan of action - Implement the solution - Verify functionality - Document findings
Example Assignments / Labs	- Troubleshooting: In this project, students will learn more about each step of the troubleshooting methodology and use these steps to repair and improve faulty network systems. - Poor Signal Strength - Interference

Unit 6: Javascript & Graphics (2 weeks, 10 hours)
This unit Introduces students 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/course/16725/explore/module/23037

Objectives / Topics Covered	- Variables - User Input - Arithmetic Expressions - Graphics - Image Representation
Assignments / Labs	- Using variables and getting user input using JavaScript - 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 - 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 7: Graphics Challenges - optional (2 weeks, 10 hours)
In this unit, students learn how to pair program! They will 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/course/16725/explore/module/23038

Objectives / Topics Covered	● Pair programming ● Solving large and more complex problems that incorporate graphics and user input
Assignments / Labs	● Several graphics challenges to tie everything learned in the module together 　○ 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 8: JavaScript Control Structures (4 weeks, 20 hours)
Learn how to use control structures such as if/else statements and loops to make more advanced programs..

Browse the full content of this unit at https://codehs.com/course/16725/explore/module/23039

Objectives / Topics Covered	● Booleans ● For Loops ● Conditionals ● Nested Control Structures ● While Loops
Assignments / Labs	● Using comparison and logical operators to control the flow of the program 　○ 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 　○ 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 　○ 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 it 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 9: Control Structures Challenges - optional (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/course/16725/explore/module/23040

Objectives / Topics Covered	● Solving large and more complex problems that incorporate control structures
Assignments / Labs	● Several challenges to tie everything learned in the module together 　　○ 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 10: Functions, Events and Lists (4 weeks, 20 hours)
Learn to write reusable code with functions and parameters.

Browse the full content of this unit at https://codehs.com/course/16725/explore/module/23041

Objectives / Topics Covered	● Functions with and without parameters ● Functions with and without return values ● Nested Control Structures ● Local variables and scope
Assignments / Labs	● Using various kinds of functions such as functions with and without parameters, and functions with and without return values 　　○ 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 11: Final Project (2-4 weeks, 10-20 hours)

This unit introduces students to the theory and practice of user interface design. Students learn about what makes an engaging and accessible user interface, and will employ an iterative design process including rapid prototyping and user testing to design and develop their own engaging program.

Browse the full content of this unit at https://codehs.com/course/16725/explore/module/23042

Objectives / Topics Covered	● Project Planning ● Final Project Creation
Assignments / Labs	● Program Requirements: ○ must use JavaScript Graphics to tell a story or give information about an important topic ○ Look back at the PSA project you completed earlier in the course if you need some inspiration! ○ must allow the user to interact with your project with their mouse ○ must use at least one timer ○ must break down the program into multiple functions ○ must utilize control structures where applicable ● Alternative Storytelling ○ Linear story ■ use a timer to switch scenes at a specific interval ■ utilize a button to switch scenes when the user decides to move on ● Choose-your-own-Adventure ○ the user is given a selection and the story plays out based on the option they choose ● Moving Scenes ○ use a timer to have different parts of your scene move around the screen to add to the user experience

Unit 12: Topics in Computer Science (1 week, 8 hours)

This unit introduces students to individual's contributions to the development of the computer and future developments of technology. Students explore applications of technology in Precision Agriculture, Artificial Intelligence, and other careers available in computer science.

Browse the full content of this unit at https://codehs.com/course/16725/explore/module/23043

Objectives / Topics Covered	● Invention of the Computer ● Precision Agriculture ● Future of Computing ● Computer Science Careers ● Inclusive Coding
Assignments / Labs	● Precision Agriculture: ○ What is Precision Agriculture ○ Drones and the Future of Farming ○ GPS Based Applications

	○ Research: The Role of Precision Technology
	● Emerging Technologies
	○ Using DNA for Storage
	○ Pros and Cons of AI
	○ AI: Is It a Bad Thing?
	● Computer Science Careers
	○ Computer Science All Around Us
	○ CS Careers
	○ Coding in the Wild
	○ Inclusive Coding

Supplementary Unit Guide:

These units can be used during the course for added practice or after the course has been completed for further review.

Supplementary Unit	Prerequisite/Recommended Unit(s)	# of activities
Extra Karel Practice	Programming with Karel Karel Challenges	12
Extra Karel Puzzles	Programming with Karel Karel Challenges	11
Functions and Parameters Practice	Functions, Events and Lists	8
Animation and Games - Extended User Interaction - *Mouse and Key Events* - *Drawing Lines* - *Crazy Ball Game*	Functions, Events and Lists	26
Extended Array Concepts - *Indexing Into an Array* - *Finding an Element* - *Removing an Element*	Functions, Events and Lists	12
Extended Data Structures - *Objects/Maps* - *Iterating Over an Object* - *Sets and Grids* - *Looping over a Grid*	Functions, Events and Lists	40