



CodeHS

Texas Fundamentals of Computer Science Course Syllabus (140 - 160 contact hours)

Course Overview and Goals

This year-long course is intended as a first course for those students just beginning their study of computer science. Students will learn about the computing tools that are used every day, while developing their ability to creatively solve real-world problems. This course introduces the basics of programming with Karel the Dog, the foundations of designing a web page with HTML/CSS, and how information is represented digitally and sent over the internet.

With a unique focus on creativity, reasoning, and applied learning, Texas Fundamentals of Computer Science gives students the opportunity to explore several important topics of computing that will foster further endeavors in the field.

Learning Environment: The course utilizes a blended classroom approach. The content is a mix of web-based and physical activities. Students will write and run code in the browser, create websites and digital presentations, and engage in in-person collaborative exercises with classmates. Teachers utilize tools and resources provided by CodeHS to leverage time in the classroom and give focused 1-on-1 attention to students.

Programming Environment: Students write and run programs in the browser using the CodeHS online editor. Students will be able to write both text based and block based programs in Karel. Students will also create several web pages using HTML and CSS. These webpages will be hosted on the CodeHS website so that they can keep a running portfolio of their creative projects, and easily share their programs with the world. Students gain programming experience early on in the course that will enable them to explore the rest of the course topics through computational thinking practices.

Quizzes: Each lesson includes at least one formative short multiple choice quiz. At the end of each unit, students take a summative multiple choice unit quiz that assesses their knowledge of the concepts covered in the unit.

Prerequisites: Texas Fundamentals of Computer Science 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 those new to computer science.

More information: Browse the content of this course at <https://codehs.com/course/21202>

Course Breakdown

Note: Most of these modules are standalone modules (with the exception that Karel Paint! and Basic JavaScript should follow Intro to Karel), which means you can change the order of the modules or break them up as you see fit.

Module 1: Welcome! (30 minutes)

Objectives / Topics Covered	<ul style="list-style-type: none"> • Why should students learn to program?
Assignments / Labs	<ul style="list-style-type: none"> • Watch a funny and inspiring TEDx video about the importance of learning to program • Reflect on interests and motivations

Module 2: Introduction to Programming with Karel the Dog (6 - 8 weeks / 30 - 40 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> • Commands • Defining versus Calling Functions • Designing methods • Control flow • Looping • Conditionals • Commenting code • Preconditions and Postconditions • Top Down Design
Example Assignments / Labs	<ul style="list-style-type: none"> • Commands <ul style="list-style-type: none"> ○ 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. • Functions <ul style="list-style-type: none"> ○ Teach Karel new commands like <code>turnRight()</code> or <code>makePancakes()</code> ○ 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 function called <code>makePancakes()</code> to help Karel solve this problem. • Top Down Design <ul style="list-style-type: none"> ○ Solve large Karel problems by breaking them down

	<p>into smaller, more manageable problems</p> <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. ● Loops and Conditionals <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. ○ 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.
--	---

Module 3: Project: Pair-Programming with Karel Paint! (1 week / 3 - 5 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Pair-programming best practices ● Project planning stages ● Top-down design ● Karel programming
Assignments / Labs	<ul style="list-style-type: none"> ● Students learn to collaborate while designing a project to completion <ul style="list-style-type: none"> ○ Stages include: setting norms, brainstorming, planning, documenting, designing, and reflecting ● Students create from scratch a complex Karel program that can be shared in their digital portfolio

Module 4: What is Computing? (2 weeks / 10 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● History of computers ● What is a computer? ● What are software and operating systems? ● What is hardware? ● Future of computing
Example Assignments / Labs	<ul style="list-style-type: none"> ● History <ul style="list-style-type: none"> ○ Find out when the first computers were created

	<ul style="list-style-type: none"> ○ Research famous computer innovators ○ What roles do computers play in your life? ○ Example Activity: <ul style="list-style-type: none"> ■ Summarize an era of advances in computers ● What is a computer? <ul style="list-style-type: none"> ○ What parts do modern computers have? ○ What are input devices? ○ What are output devices? ○ Example Activity: <ul style="list-style-type: none"> ■ Draw a computer and label all of its parts, including the input devices and output devices ● Software/Hardware <ul style="list-style-type: none"> ○ What's the difference? ○ What hardware components make up a computer? ○ What is software used for? ○ Example Activity: <ul style="list-style-type: none"> ■ Label the parts of your computer ● Operating Systems <ul style="list-style-type: none"> ○ What are they for? ○ What different versions are there? ○ Example Activity: <ul style="list-style-type: none"> ■ Run simulations and compare ● Future of Computing <ul style="list-style-type: none"> ○ Research uses of Artificial Intelligence in use now ○ Research new ways of storing data ○ Example Class Activity: <ul style="list-style-type: none"> ■ In what ways can we use technology that we couldn't 10 years ago. Are these technological advances helpful or harmful overall?
--	---

Module 5: Digital Information (2 weeks / 10 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● How digital data is represented ● Encoding data ● Converting and using different number systems <ul style="list-style-type: none"> ○ Binary ○ Hexadecimal ● Manipulating images at the pixel level
Example Assignments / Labs	<ul style="list-style-type: none"> ● Encoding data <ul style="list-style-type: none"> ○ Create your own encoding scheme ○ Encode images using binary ○ Example Activity: <ul style="list-style-type: none"> ■ Write a message by encoding the characters in binary, using the ASCII codes. ● Using different number systems

	<ul style="list-style-type: none"> ○ Convert numbers between decimal, binary, and hexadecimal ○ Example Activity: <ul style="list-style-type: none"> ■ Earn a high score playing the decimal to binary game or decimal to hexadecimal game. Click on the digits to change their values and make the binary or hexadecimal number match the target decimal value. ● Manipulating Images <ul style="list-style-type: none"> ○ Make different colors by changing the amount of red, green, and blue present ○ Create image filters ○ Create images pixel by pixel ○ Example Activity: <ul style="list-style-type: none"> ■ Create a pixel rainbow by typing out the hexadecimal color encoding for each pixel
--	--

Module 6: Basic Javascript (2 - 3 weeks / 10 - 15 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Computer Science Careers ● Variables ● User Input ● Arithmetic Expressions ● Random Numbers
Assignments / Labs	<ul style="list-style-type: none"> ● Learning about a variety of careers in the field of computer science. ● 1-3 exercises per topic for a total of 10 exercises. ● 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! ● Computing mathematical operations ● Using a random number generator <ul style="list-style-type: none"> ○ Example Exercise: building a program that tests the users on their multiplication tables

Module 7: The Internet (2 weeks / 10 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Structure of the internet ● How network data is transmitted ● Hardware involved in the transmission of data
-----------------------------	---

	<ul style="list-style-type: none"> ● How the internet has impacted everyday life
Example Assignments / Labs	<ul style="list-style-type: none"> ● Structure of the internet <ul style="list-style-type: none"> ○ Explore the differences between IPv4 and IPv6. Why are we running out of addresses? ○ Explore the different levels of the internet. ○ Example Activity <ul style="list-style-type: none"> ■ Trace a website request from the server, through the network, and to your computer ● How data is transmitted <ul style="list-style-type: none"> ○ 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: <ul style="list-style-type: none"> ■ 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 <ul style="list-style-type: none"> ○ 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 8: Web Design (6 -7 weeks / 30 - 35 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Creating webpages using HTML <ul style="list-style-type: none"> ○ Links ○ Images ○ Lists ○ Tables ○ Inline styling ● Styling webpages with CSS <ul style="list-style-type: none"> ○ Creating CSS rules ○ CSS classes ○ CSS IDs ● Embedding script into HTML ● How webpages are requested and delivered
Example Assignments / Labs	<ul style="list-style-type: none"> ● Links <ul style="list-style-type: none"> ○ Learn how to link different pages together ○ Example Exercise: Create a webpage that provides links to your favorite books ● Images

	<ul style="list-style-type: none"> ○ Learn how to add and format images ○ Example Exercise: Create a collage of images ● Tables <ul style="list-style-type: none"> ○ Learn how to create and style tables ○ Example Exercise: Create a table describing your favorite music artists and songs ● Styling with CSS <ul style="list-style-type: none"> ○ Use CSS to add background colors, font colors, font styles, borders, and position elements on the page ○ Example Exercise: Create CSS classes and IDs to apply formatting to a BINGO board ○ Example Exercise: Create CSS classes to style a music library web page ○ Example Exercise: Create CSS Rules to put a Karel puzzle together ● Viewing websites <ul style="list-style-type: none"> ○ How are web pages served to your computer? ○ Where do web pages live? ● Final Project <ul style="list-style-type: none"> ○ Build your own homepage using everything you've learned in the module ○ This homepage will serve as your personal portfolio of creative projects as you continue through the course!
--	---

Module 9: Project: Designing for Impact (2 - 3 weeks / 10 - 15 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Prototyping ● Testing ● Project Planning ● Final Project Creation
Assignments / Labs	<ul style="list-style-type: none"> ● Collaborative open-ended final project which encourages creativity ● Requirements <p>The list below lays out the minimum requirements of your website. Feel free to go big and add even more!</p> <p>Your website:</p> <ul style="list-style-type: none"> ■ must detail the community issue you are addressing ■ must include at least 3 pages covering the following: <ul style="list-style-type: none"> ● general information, including contact information and a link to the chosen organization's website ● background of the issue, including an

	<p>interactive data visualization showing an important statistic</p> <ul style="list-style-type: none"> ● how the chosen organization is working to find a solution and why they should be sponsored by NuWorld ■ must include a new logo that morphs NuWorld's logo with that of the chosen organization (NuWorld's logo will be provided in a future lesson). This new logo can either be in a static image format or a gif.
--	--

Module 10: Digital Citizenship and Cyber Hygiene (3 - 4 weeks / 15 - 20 hours)

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Digital Footprint and Reputation ● Cyberbullying ● Internet Safety ● Privacy and Security ● Information Literacy ● Creative Credit and Copyright ● Hacking Ethics
Example Assignments / Labs	<ul style="list-style-type: none"> ● Digital Footprint and Reputation <ul style="list-style-type: none"> ○ 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: <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? ● Cyberbullying <ul style="list-style-type: none"> ○ 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: <ul style="list-style-type: none"> ■ Explore cyberbullying scenarios: What would you do? ● Internet Safety

	<ul style="list-style-type: none"> ○ What are some ways to stay safe online? ○ What are some online safety guidelines? ○ Example activities: <ul style="list-style-type: none"> ■ Explore Internet safety scenarios: What would you do? ● Privacy and Security <ul style="list-style-type: none"> ○ 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: <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"> ○ What is information literacy? ○ How can you do effective internet searches? ○ What are some techniques for judging source legitimacy and identifying misinformation? ○ Example activities: <ul style="list-style-type: none"> ■ Create and test search queries ■ Explore evidence for using sources ● Creative Credit and Copyright <ul style="list-style-type: none"> ○ What is copyright? ○ What are the different types of copyright licenses? ○ Example activities: <ul style="list-style-type: none"> ■ Create citations for sources ■ Explore image search tools ● Hacking Ethics <ul style="list-style-type: none"> ○ What are hackers? ○ Are there different kinds of hackers? (white, black, grey) ○ What are bug bounty programs? ○ Is hacking always illegal? ○ What are the consequences of illegal hacking? ○ Example activities: <ul style="list-style-type: none"> ■ Explore what penetration testing is ■ Sign ethical hacker agreement
--	---

Unit 11: Computer Science Careers (2-4 weeks/10-20 hours)

Students learn about a variety of computer science careers and organizations, and what the next steps could look like for them if interested.

Objectives / Topics Covered	<ul style="list-style-type: none"> ● Careers and internships ● CS career preparation ● Legal and ethical responsibilities
-----------------------------	--

	<ul style="list-style-type: none"> • Workplace readiness
Example Assignments / Labs	<ul style="list-style-type: none"> • Exploring computer science careers, internships, and organizations • Learning about CS resumes and certifications • Researching about a major ethical or legal topic in CS • Reflecting on what it means to be a leader and the skills required to be successful in the workplace

Optional Supplemental Materials

Supplementary Units	Prerequisite/Recommended Unit(s)	# of activities
Coding in Sports	None. Introduction to coding via block coding	11
Extra Karel Practice	Introduction to Programming	11
Extra Karel Puzzles	Introduction to Programming	10
What is Cybersecurity?	None	31