Nebraska Introduction to Computer Science and Technology: Coding First Syllabus

High School (60-70 Contact Hours)

Course Overview and Goals

This introductory course develops students' abilities to analyze, evaluate, and reflect upon technologies such as computer hardware, computer software, networking, and security. Students will learn the fundamentals of programming, build computational thinking skills, and reflect on the impact of computing on society.

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 are made up of short video tutorials, example programs, quizzes, simulations, programming exercises, and free-response prompts. Each unit ends with a comprehensive unit quiz that assesses students' mastery of that unit's material.

Programming Environment

Students write and run programs in the browser using the CodeHS editor. The default programming module is in Python and utilizes Karel the Dog (coding first) and Tracy the Turtle (coding last) to teach basic programming concepts. There are two other coding options (Python and JavaScript) in the supplemental modules that can be swapped out and used instead. Only one programming module is necessary to hit all required standards.

More Information

Browse the content of this course at https://codehs.com/course/26416/explore

Technology Requirements

To complete all activities and exercises in this course, students must have access to the 3rd party sites and tools listed here: NE Introduction to Computer Science Links

Prerequisites

The Nebraska Introduction to Computer Science and Technology: Coding First 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.

Course Breakdown

Module 1: Coding with Karel (3-4 weeks/15-20 hours)

In this module, students learn the basics of programming by giving commands to a computer just like you give commands to a dog. Karel is a dog that lives in a grid world and can be instructed to move around and pick up

and put down tennis balls. Students will learn Python commands, functions, and control structures by solving puzzles and writing creative programs for Karel to follow. *This module can be used in addition to, or in place of, the Programming with Turtle Graphics module.*

Browse the full content of this unit at https://codehs.com/course/26416/explore/module/37382

Objectives / Topics Covered	• Commands • Defining vs. Calling Functions • Designing Functions • Program Entry Points • Control Flow • Looping • Conditionals • Commenting Code • Preconditions and Postconditions • Top Down Design

Assignments / Labs	• 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 columns. Each stack of pancakes should have three pancakes. Create a function called `makePancakes()` to help Karel solve this problem. The world should end up exactly as shown here.
	  ○
	• 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 row, jumping over all of the hurdles. Karel should only jump if there is a hurdle blocking the way. However, the hurdles can be in random locations. The world is fourteen columns long. You must write a function named `jumpHurdle()` as part of your solution.

Module 2: Cybersecurity and You (2 weeks/10 hours)

In this module, students delve into key areas such as personal data collection, the reliability of online information, cyber ethics and laws, personal data security, cybersecurity essentials, and strategies to combat common cyber threats and their prevention, equipping individuals with the knowledge to navigate the digital landscape responsibly and securely.

Browse the full content of this unit at https://codehs.com/course/26416/explore/module/37383

Topics Covered	● Digital Footprint and Responsibility ● Personal Data Collection and Security ● Cyber Ethics and Laws ● Cybersecurity Essentials ● Common Cyber Attacks and Prevention
Example Assignments	● Digital Footprint and Responsibility ○ Students explore the impact of social media and technology on teenagers, covering topics like digital footprints, the rise of social media screenings, cyberbullying, and the importance of updating privacy settings. ● Personal Data Collection and Security ○ This lesson delves into the use and security of personal data, discussing how companies like Google utilize user information, the implications of location tracking, legal aspects of privacy, and encourages critical thinking through reflections, checks for understanding, and explorations of browser security settings and the trade-offs of security measures. ● Cyber Ethics and Laws ○ This lesson navigates through cyber ethics, differentiating between ethics and laws, exploring legal consequences, copyright in education, the process of obtaining permissions, and the pros and cons of intellectual property laws. ● Cybersecurity Essentials ○ This lesson covers cybersecurity, featuring activities on the AAA Security Framework and the CIA Triad, along with exploring the impact of the Internet of Things on data security.

Module 3: Cryptography (1 weeks/3 hours)

In this module, students will dive into the history of cryptography systems, the motivation behind using encryption systems, and modern cryptography systems. This includes explaining the core concepts of Public Key Infrastructure and hash functions. Students will also explore the importance of digital certificates, and authentication methods.

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37384

Topics Covered	● Cryptography: Then, Now and Future ● Symmetric Encryption ● Asymmetric Encryption ● Public Key Encryption ● Authentication Methods ● Digital Certificates
Example Assignments	● Symmetric Encryption ○ *Rail Fence Cipher:* The Rail Fence Cipher is a form of transposition cipher that uses columns and rows. The plaintext is written downwards and bounces back and forth on a diagonal. The 'rails' refer to the number of rows. Decrypt the message below using 5

	rails. Using the Rail Fence Cipher, encrypt your own message and trade with a partner. See if you can decrypt the message without knowing how many rails your partner used. How could you make this cipher even stronger? ● Asymmetric Encryption ○ *Public Key Encryption Exploration*: You would like to send a message to your friend. Your friend will need a private key as well to add to the encryption. Complete the chart according to the rules of the public key and the two private keys. ● Authentication Methods ○ *Guess The Hash:* Work with a partner and take turns hiding a password, and trying to guess the password. Without letting your partner see, type in a simple word (about 3-7 letters long) as your password. You'll see the corresponding hash in the output box. Using the scrambled letters, attempt to guess the password. You will only know if you have guessed the password if the hashes shown match.

Module 4: System Administration (2 weeks/10 hours)

In this module, students will compare and contrast common operating systems (Windows, Linux, OS) and explain the importance of application security. They will investigate security options and implement user accounts to enforce authentication and authorization. Students will also demonstrate how to work with basic and advanced command prompts.

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37385

Topics Covered	● Operating Systems ● Software and Applications ● Application Security ● Browser Configuration ● System Administration ● Command Line Interface
Example Assignments	● Understanding Operating Systems ● Comparing Operating Systems ○ Installing an OS ● File Management ○ What Processor are you Running? ● Software Licenses ● Antivirus Software ○ Data Backups ● Using Cache ● Popup Blockers ● User Accounts ○ Admin vs. Standard ● Host Security ○ Using a Log ● System Commands ○ cd, ls, mk etc ● Network Commands ○ ipconfig, netstat etc

Module 5: IT Concepts (2 weeks/7 hours)

In this module, students explore the structure and design of the internet and networks, and how this design affects the reliability of network communication, the security of data, and personal privacy. Students will learn how the Internet connects computers all over the world by use of networking protocols.

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37386

Topics Covered	• Binary • Encoding Text and Images in Binary • IP Addresses • Routing and Packets • Protocols: TCP, UDP, HTTP/HTTPS • How do Websites Work? • OSI Model • Impact of the Internet
Example Assignments	• Write a Message in Binary • Create a Color Pixel Image • Routing with ARPANET • Journey of a Web Page • Troubleshooting with the OSI Model • Compass Points: The Internet ○ In this activity, students use the Compass Points thinking routine to examine your feelings about the internet and its impact on society.

Module 6: IT Infrastructure (2 weeks/6 hours)

In this module, Students will learn about the physical elements of computers and networking such as motherboards, RAM, routers, and the use of port numbers, ethernet, and wireless devices.

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37387

Topics Covered	• Internal Components of a Computer • Peripheral Devices • Network Devices • Storage and Network Options • Network Communication • Network Management

Example Assignments	• Different Types of CPU
	• RAM vs. Hard Drive
	• Wireless Internet Connections
	  ○ Speed Test
	• Security of Cloud Storage
	• Ethernet Standards
	• Setting Up a Firewall
	  ○ Establish Firewall Rules
	• SSH Logs
	  ○ Reading Logs

Module 7: Programming with Turtle Graphics (3 weeks/12 hours)

In this module, students learn the basics of programming using Python and Tracy the Turtle. They learn Python commands, functions, control structures, and user interaction by solving puzzles and writing creative programs for Tracy to follow. *This module can be used in addition to, or in place of, the Coding with Karel module.*

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37388

Topics Covered	• What is a Command? • Moving Tracy • Tracy's Coordinate System • For Loops • Functions and Parameters • Top Down Design • Abstraction • Variables • User Input • If/Else Statements • While Loops

Example Assignments	• Row of Circles
	  ○ In this program, students will have Tracy draw a row of circles across the width of the canvas using a for loop.
	• Bubble Wrap 2.0
	  ○ In this program, students will have Tracy add highlights to each bubble from our Bubble Wrap example program. They will use top down design to break this large problem into smaller pieces!
	• Beaded Bracelet
	  ○ In this program, students will have Tracy create a beaded bracelet using functions and circles.
	• Four Colored Triangles
	  ○ In this program, students will have Tracy draw four tri-colored triangles next to one another in the center of the canvas.

Supplemental Module: Programming in Python (3 weeks/15 hours)

Students learn the fundamentals of programming in Python by running example code and completing relevant coding exercises. *This module can be used in place of Karel the Dog or Tracy the Turtle.*

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37562

Topics Covered	• Introduction to Python • Printing in Python • Variables and Types • User Input • Mathematical Operators • String Operators

Example Assignments	• Printing in Python
	  ○ Students write a program that prints their name and something about themselves.
	• Variables and Types
	  ○ Write a program that does the following:
	  ■ Creates a string variable.
	  ■ Creates an integer variable.
	  ■ Prints both variables, each on its own line.
	• User Input
	  ○ Write a program that takes a user's name and says hello to them. Save their name in a variable.

Supplemental Module: Programming in JavaScript (3 weeks/15 hours)

Students learn the fundamentals of programming in JavaScript by running example code and completing relevant coding exercises. *This module can be used in place of Karel the Dog or Tracy the Turtle.*

Browse the full content of this module at https://codehs.com/course/26416/explore/module/37563

Topics Covered	• Variables • User Input • Random Numbers • Functions • If/Else Statements • Logical & Comparison Operators • While Loops • For Loops

Example Assignments	• 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!
	• 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.
	• Using for loops
	  ○ Example Exercise: Chalkboard
	  ■ You have just gotten in trouble in school for coming late. The punishment is that you have to write "I will not come late to school" 100 times on the board. That would take you a while to do... but luckily you have for loops. Using a for loop, print "I will not come late to school" 100 times.