Florida Computer Programming Fundamentals Honors

High School (170-175 Contact Hours)

Course Overview and Goals

The Florida Computer Programming Fundamentals Honors course offers an introduction to the essential concepts of computer science through programming. Students will explore key topics such as problem-solving techniques, data structure organization for managing large datasets and the development and implementation of algorithms for data processing and information discovery.

The course emphasizes the evaluation of potential solutions and considers the ethical and social impacts of computing systems. A strong focus is placed on object-oriented programming and design using an appropriate programming language. This course covers core programming concepts typically found in a first-semester college-level Computer Science course, preparing students for more advanced studies in the field.

Learning Environment

The course utilizes a blended classroom approach. The content is a mix of web-based and physical activities. Each module of the course is broken down into lessons. Lessons are composed of short video tutorials, interactive learning pages, quizzes, explorations, simulations, and free-response prompts. Each module ends with a comprehensive quiz that assesses students' mastery of that module's material.

More Information

Browse the content of this course at https://codehs.com/course/26482/overview

Prerequisites

The Florida Computer Programming Fundamentals Honors 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. A solid understanding of mathematical reasoning, including linear functions and the Cartesian (x, y) coordinate system, is recommended for success in this course.

Technology Requirements

To complete all activities and exercises in this course, students must have access to the 3rd party sites and tools listed here: Florida Computer Programming Fundamentals Honors Course Links

Course Breakdown

Module 1: Cybersecurity and You (3 weeks/15 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.

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, and 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 2: IT Concepts (2 weeks/10 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.

Topics Covered	• IP Addresses • Routing and Packets • Protocols: TCP, UDP, HTTP/HTTPS • How do Websites Work? • Impact of the Internet

Example Assignments	• Submarine Cable Map Exploration
	  ○ This handout introduces students to the critical infrastructure of submarine cables that power the global internet. Through a combination of video-based learning and interactive map exploration, students investigate real-world submarine cables, analyze their physical characteristics, ownership, and international connections, and reflect on their impact and future alternatives such as satellite internet.
	• Impact of the Internet
	  ○ *Compass Points: The Internet:* In this activity, students use the Compass Points thinking routine to examine their feelings about the Internet and its impact on society.

Module 3: IT Infrastructure (3 weeks/15 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.

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

Example Assignments	• Network Devices
	  ○ *Network Troubleshooting:* Jamal's computer is able to connect to the Wi-Fi signal, but there is no Internet access. Which device do you think might be causing the problem and why?
	• Network Options
	  ○ *Wireless Network Setup:* In this activity, students will draw a diagram that represents a wireless network setup that will be implemented for a fictitious house, office, or apartment building. The teacher will either assign them a building or they can create one from their own imagination.
	• Network Management
	  ○ *SSH Logs:* Addison works as a server administrator and has been accused of stealing company financial data. He swears he is innocent. A search warrant has been granted for the company's network logs and you have been tasked with learning as much as possible about the attack and the attacker. Can you dig into the logs and help track down the hacker?

Module 4: Programming in Python (3 weeks/15 hours)

In this module, students are introduced to the fundamentals of programming by learning how to write basic code in Python using print statements, variables, user input, and arithmetic expressions. They'll explore data types, string operations, comments, and the role of programming languages in creating interactive programs.

Objectives / Topics Covered	• Printing • Variables • Types • User Input • Converting Input Types • Arithmetic Expressions • String Operators • Comments • Programming Languages

Example Assignments	• Printing
	  ○ Print messages to the console
	• Variables
	  ○ Create variables of different types, and print them to the console.
	• Types
	  ○ Investigate the types of different variables
	  ○ Convert between types

	• Arithmetic Expressions & Converting Input Types
	  ○ Age in One Year - Ask the user how old they are, and tell them how old they will be in one year
	  ○ Rectangle, part 1 - Make variables for length and width and compute area and perimeter
	  ○ Rectangle, part 2 - Ask the user for length and width, and compute area and perimeter

Module 5: Decisions in Programming (2 weeks/10 hours)

In this module, students expand their programming skills by working with Boolean values, logical and comparison operators, and if statements to create decision-making programs. They apply their understanding to secure coding practices and explore how programming decisions align with real-world cybersecurity policies and principles.

Objectives / Topics Covered	• If Statements • Boolean Values • Logical Operators • Comparison Operators • Floating Point Numbers

Example Assignments	• If statements and boolean values
	  ○ Is it raining? - Write a program that uses a boolean variable to determine whether or not it is raining
	• Boolean operators and expressions
	  ○ Boolean variable - Take a variable and use it in an if statement
	  ○ Legally allowed to vote - User reports age and the program tells them whether or not they can vote in the US
	  ○ Transaction - The user reports the balance and deposit/withdrawal, and the program prints a new balance or error
	  ○ Recipe - Ask the user for ingredients, amounts per serving, and number of servings, and report the total amount of each ingredient needed

Module 6: Looping (2 weeks/10 hours)

Students learn how to write more efficient code by using loops as shortcuts.

Objectives / Topics Covered	• While Loops • For Loops • Break and Continue • Nested Control Structures

Example Assignments	• While Loops
	  ○ Divisibility - Ask the user to enter a numerator and denominator, and re-prompt until the denominator is non-zero
	• For Loops
	  ○ Average test score - Compute the average of several test scores
	• Break and Continue
	  ○ Higher/ Lower - Ask the user to guess a particular number between 1 and 100. If the user's guess was too high or too low, they should be notified
	• Nested Control Structures

	○ Rolling Dice - Print out all combinations that can be made when 2 dice are rolled

Module 7: Project: Password Authenticator (3 days/3 hours)

Students write a program to provide feedback on whether the entered password is correct or incorrect.

Objectives / Topics Covered	● Variables ● User Input ● Boolean Operations ● Loops ● Control Structures
Project Description	● Create a password authentication program that prompts the user to enter a password. The program should compare the entered password with a predefined correct password and provide feedback on whether the entered password is correct or incorrect. The program should allow the user to try again a limited number of times before locking them out.

Module 8: Functions and Exceptions (2 weeks/10 hours)

Students learn how to decompose problems into smaller pieces that work together to solve a problem.

Objectives / Topics Covered	● Functions ● Namespaces ● Parameters ● Return Values ● Exceptions
Example Assignments	● Example exercises: ○ Functions ■ Raining cats and dogs - Write functions to print text art of a cat and a dog ■ Temperature converter - write functions to convert from Fahrenheit to Celsius and vice versa ○ Exceptions ■ Temperature converter, part 2 - Add exception handling to your temperature conversion program ○ Putting it all together ■ Enter a positive number - Make a function to repeatedly ask the user to enter a number until they enter a positive number

Module 9: Strings and Data Structures (3 weeks/15 hours)

Students learn more sophisticated strategies for manipulating text in their programs. Students also learn how tuples and lists are formed and the various methods that can alter them.

Objectives / Topics Covered	● Indexing and Slicing ● Math Operators on Strings ● For Loops Over a String ● String Methods ● Tuples ● Lists ● For Loops and Lists

	• List Methods

Example Assignments	• Example exercises:
	  ○ Indexing
	  ■ First character - write a function that takes a string and returns the first character
	  ■ All but the first character - write a function that takes a string and returns everything but the first character
	  ○ Math operators and strings
	  ■ Full name - write a function that takes two strings (a first name and a last name) and returns a full name as a single string
	  ■ Replace a letter - write a function that takes a string and returns a copy with the character at a particular index replaced with a dash
	  ○ For loops on strings
	  ■ Count occurrences - write a function that takes two strings and returns the number of times the second string appears in the first string
	  ○ String methods
	  ■ Add enthusiasm - write a function that takes a string and returns that string in all upper case
	  ■ Remove all from string - write a function that takes two strings and returns a string that consists of the first string with all instances of the second string removed
	  ○ Tuples
	  ■ Cookout Orders - Given a tuple of food orders, add up the number of burgers and hotdogs and print the total sums.
	  ○ Lists
	  ■ Listed Greeting - Ask a user to enter their name, age, and favorite sport, then split their response into list elements and use index values to greet them by name and respond that you enjoy that sport as well!
	  ■ Exclamat!on Po!nts - Ask the user for a string and then print the same string with every lowercase i replaced with an exclamation point.
	  ■ Librarian - Ask the user for the last names of the authors of the five books they are returning. Print a list of those names in sorted order.

Module 10: Classes and Objects (2 weeks/10 hours)

Students learn about classes and the principles of object-oriented design.

Objectives / Topics Covered	• Classes and Objects • Methods • Built-In Methods • Operator Overloading • Class Variables vs. Instance Variables • Inheritance • Hidden Attributes • Namespaces • Modules

	• Computational Limits

Example Assignments	• The Rectangle Class
	  ○ In this assignment, you will create an instance of a class, set length and width variables, and print the length and width variables for a rectangle.
	• Contact Merge
	  ○ In this exercise, you will get some practice with the __add__ method by implementing it for a class called ContactBook. This class represents a collection of names and phone numbers.
	• Food, Part 1 and 2
	  ○ In Part 1, you'll create a class hierarchy with Food, Vegetable, and Broccoli, where each class inherits from the one above and sets specific instance variables using constructor chaining. In Part 2, you'll write a loop that checks if each item in a list is a Food object and prints its details or a message if it's not.
	• Exploration: The Halting Problem
	  ○ In this activity, you'll explore a deceptively simple function that highlights a deep computer science concept, the Halting Problem, by predicting and testing whether certain inputs will cause the program to stop or run forever.

Module 11: Roles in a Software Development Team (1 week/5 hours)

Students learn the key roles and responsibilities of members of a software development team.

Objectives / Topics Covered	• Software Engineers • Quality Assurance Engineers • Designers • Project Managers

Example Assignments	• Create a Mood Board
	  ○ In this assignment, you will act as a designer and create a mood board for a store of your choosing. To visually represent the brand and theme of the store, your mood board must include the following:
	  ■ 1. A color palette that best represents the store's brand
	  ■ 2. One or two fonts that align with the store's identity
	  ■ 3. Images related to the store's products or target audience
	• Create a Task Board
	  ○ Imagine that you are a Project Manager. Before assigning work to members of the software development team, you need to create a list of tasks needed to create an application for the store you created a mood board for in the previous lesson.

Module 12: Creating Webpages (2-3 weeks/10-15 hours)

In this module, students learn HTML and CSS basics as they build simple webpages.

Objectives / Topics Covered	• Introduction to HTML • Structure of an HTML Page

	• Formatting Text • Introduction to CSS • HTML and CSS • Multi-file Websites • Embedding iframes • Using the Inspector • Image Manipulation • Animation • Interaction • Software Development Life Cycle • Project: Create Your Homepage

Example Assignments	• Button Interaction
	  ○ In this activity, you will Use CSS to style buttons so that "clickable" ones change color on hover and click, while "unclickable" ones stay gray and show a disabled cursor using cursor: not-allowed.
	• Using the Inspector and Mobile View
	  ○ In this activity, you will use your browser's Inspector and mobile view tools to test and adjust a webpage for mobile devices, making sure fonts, buttons, images, and layout all display and function correctly on different screen sizes.
	• Create Your Homepage
	  ○ In this activity, you will create your own personal homepage at <your username>.codehs.me that includes your name, bio, profile image, portfolio links, an embedded Python project, custom CSS styles, and a mobile-friendly layout to showcase your work from this course.

Module 13: The Data Science Life Cycle (4 weeks/20 hours)

Students will learn and apply the process of the data science life cycle. This includes asking statistical questions, collecting or obtaining reliable raw data, analyzing the data using measures of central tendency and spread and interpreting and summarizing the results.

Objectives / Topics Covered	• What is Data Science?
	• Gathering Data
	  ○ Quantitative/Qualitative
	• Exploring Data Using Python
	• Modules and Libraries
	• Using the Pandas Library
	  ○ Series
	  ■ Measures of Central Tendency
	  ■ Measures of Spread
	  ○ DataFrames
	  ■ Selecting Columns
	  ■ Using Functions

Example Assignments	• Mini-Project: Students will go through the first two steps of the data cycle using data of their choosing.
	  ○ Ask Questions: Formulate a statistical question that can be answered with data.
	  ○ Consider Data: Collect or find data that will aid in answering your question.
	  ○ Analyze Data: Perform statistical analysis, run calculations and/or

	create data displays to identify patterns and relationships ○ Interpret Data: Answer questions and summarize the results. ● Hot Dog Plots: Use the correct Python functions to create a boxplot of the data. Using the graph, determine the summary statistics and the spread. ● Roller Coaster Rankings: Define a function that will compute a score for each roller coaster. Use this function to store the results in a new column. ● Student Test Scores: Create a function that finds the maximum test score between test one and test two for each student. Create a function that finds the maximum test score between all three tests for each student. Decide which calculations, along with these two new columns, can help you answer the original statistical question? Explore and further analyze your data until you come to a conclusion.

Module 14: Intro to AI (2 weeks/10 hours)

In this module, students will gain an understanding of the main concepts and vocabulary around AI.

Objectives / Topics Covered	● Introduction to Artificial Intelligence ● Generative vs Predictive AI ● Large Language Models ● Prompt Engineering ● Introduction to Machine Learning
Example Assignments	● Generative vs Predictive: Explore Google Experiments ○ Students explore different applications of AI through Google Experiments ● Large Language Models: Chatbot Arena ○ Students compare LLMs through the Chatbot Arena tool ● Prompt Engineering: Generate an Image ○ Students use prompt engineering techniques to generate an image ● Introduction to ML: CNN Visualization ○ Students observe how a CNN gathers data on an image and makes a prediction as to what it could represent
AI Tools/Permissions	● Google Experiments ● Chatbot Arena ● ChatGPT/Gemini ● Hugging Face

Module 15: Project: The Engineering Design Process (2-3 weeks/12 hours)

In this project, students will learn the theory and practice of the engineering design process. This project allows students to think creatively about the applications of the concepts covered in the course and create something of personal value.

Topics Covered	● Design Thinking ● Prototyping ● Testing ● Project Prep and Development

Example Assignments	● Using Data ○ Creating a Survey ○ Data Cleaning ○ Comparing Datasets ○ Drawing Conclusions ● Prototyping and User Testing ○ Wizard of Oz Prototyping ○ How to User Test Responses