Idaho Digital Literacy Syllabus

High School (60 Contact Hours)

Course Overview and Goals

In the Idaho Digital Literacy course, students will gain a comprehensive understanding of the digital world they navigate every day. They will learn the importance of safeguarding our increasingly digital world from unauthorized access, ensuring the privacy and integrity of data. Students will also be introduced to the fundamentals of computer programming, with an emphasis on helping students develop logical thinking and problem-solving skills. Students learn to design, code, and test their programs while applying mathematical concepts.

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/26342/overview

Prerequisites

The Idaho Digital Literacy 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.

Technology Requirements

To complete all activities and exercises in this course, students must have access to the 3rd party sites and tools listed here: Idaho Digital Literacy Course Links

Course Breakdown

Module 1: Cybersecurity and You (2 weeks/10 hours)

In this module, students explore 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: Technology and Society (1.5 weeks/8 hours)

In this module, students examine how the internet and digital technologies impact various aspects of modern life, from personal behavior to global communication. Through readings, reflections, and interactive routines, learners explore both the benefits and challenges of a digitally connected world.

Topics Covered	• Impact of the Internet on Society • Ethical and Social Implications of Technology • Modeling and Designing Embedded Systems • Data as a Valuable Resource • Understanding Human vs. Artificial Intelligence

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 3: Basic Python and Console Interaction (2.5 - 3 weeks/14 hours)

Students learn the basics of programming by writing programs that interact with users through the keyboard.

Topics Covered	● Printing ● Variables ● Types ● User Input ● Converting Input Types ● Arithmetic Expressions ● String Operators ● Comments
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 4: Project: Mad Libs (2 days/2 hours)

Students write a program that allows users to create their own Mad Lib stories using variables and user input.

Topics Covered	● Variables ● User Input ● Print Statements
Example Assignments	● Create a Mad Libs game where users can input different types of words to complete a story. The program should prompt the user for various types of words (e.g., noun, adjective, verb) and then construct a story using the provided words.

Module 5: Conditionals (2 weeks/10 hours)

Students teach their programs to make decisions based on the information they receive.

Topics Covered	• If Statements • Boolean Values • Logical Operators • Comparison Operators • Floating Point Numbers and “Equality”

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 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: Project: Quiz Game (2 days/2 hours)

Students write a program that quizzes users on a series of multiple-choice questions and checks their answers.

Topics Covered	• Variables • Boolean Logic • Arithmetic Operators • Conditionals • Comparison Operators
Project Description	• Create a simple multiple-choice quiz game using Python. The program should ask the user a series of questions and check their answers. After all the questions have been answered, the program should display the final score.

Module 7: Looping (2 weeks/10 hours)

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

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 8: Project: Password Authenticator (2 days/2 hours)

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

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 9: Project: Remix a Program (2 days/2 hours)

Students will be introduced to the concept of code reuse and modification by remixing a program created by another student. They will learn about attribution and the importance of giving credit to the original creators of code.

Topics Covered	• Code Reuse • Remixing a Program • Attribution
Project Description	• In this project, you will remix another student's program! To remix a program, you take a program developed by someone else and change it by adding, removing, or altering the code. For the original program, you can share and trade projects with a student in your class, or choose a project that has been posted online in Student Projects.

Other Coding Options

Supplemental Modules	Instructions
• Turtle Graphics: Programming • Turtle Graphics: Challenges with Tracy	These two modules can be pulled into the course to replace modules 3-9. The two modules will take the same amount of time, which is approximately 9 weeks or 45 hours.

• JavaScript: Basics • JavaScript: The Canvas and Graphics • JavaScript: Graphics Challenges • JavaScript: Control Structures • JavaScript: Control Structures Challenges	These five modules can be pulled into the course to replace modules 3-9. The five modules will take the same amount of time, which is approximately 9 weeks or 45 hours.