

Computer Club Coding Problems

Overview:

This is a list of problems that help teach you how to code. The problems are specifically designed to teach you new elements of your language, make you think deeply about the logic behind them, and learn to debug and optimize your programs.

Problem completions are recorded in this google sheet. It is meant to help you track completions. Speak to a board member to get a completion.

[Computer Club Coding Problem Completions](#)

If you need a space to code in, you can use <https://www.mycompiler.io>. It is a smart IDE for popular languages. (NodeJS = Javascript)

Members' Languages:

This is a list of everyone in the club and the languages they are strong in. Speak to one of us if you need help on problems on figuring them out. If you want to be added to this list, let a board member know.

- Spencer Schraven: Python
- Dax Mikow: Java, Javascript

Resources:

If you feel like you don't have the fundamental solution to completing these problems, feel free to reference the documentation for your language or look up the solution online. Remember though, **This is meant to teach you how to code. Do not use other's answers as your own, but learn from them.**

Most languages have documentation for them. Here is the official documentation for the following languages.

- Python: <https://docs.python.org/3/>
- Javascript: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- Java: <https://docs.oracle.com/en/java/javase/17/docs/api/index.html>

Format:

Each problem will have a prompt that is meant to be completed by programming in whatever language you are comfortable in or learning. Some of them have definite solutions where math is required, while others have guidelines for making certain things.

Problems will have at least one of the following tags:

Learning: Problems meant to teach you new things and features about your language.

Logic: Problems meant to make you deeply think about the logic behind your program.

Optimization: Problems meant to teach how to cut down on the operations required to run it.

Math: Problems that use mathematics in complex and difficult ways.

Euler(#, link): Problems from <https://projecteuler.net/archives>. The number is the number problem in PE.

Problems:

1: Print *Hello world!* to your console.

Learning

2: Make a variable, then save its value as the sum of the current month and day as numbers, then subtract the last two digits of the current year, and print it to the console.

Learning, Logic

3: Create a function that prints *Running!* to the console. Call this function in your code.

Learning

4: Create a function that when called prints an input string to the console. Call this function 3 times, with *"I"*, *"Love"*, and *"Coding!"* as input strings.

Learning

5: Create a function that takes two number parameters and multiplies them, then adds both numbers to it, then returns that value. For example, if it used 4 and 6, it should return 34. Use this method to print the result of putting 3729 and 1932 into the function.

Learning, Logic

6: Print problem 5's function with 2783 and 10923 in the format of *The answer is (Answer)!*

Learning

7: Print to the console the first six digits of the number resulting from taking the square root of 563 cubed, divided by 30, then rounded to the nearest whole number.

Learning, Logic

8: Store the values *"I"*, *"love"*, *"programming"*, *"so"*, and *"much!"* in one variable, and using only the one variable, print to the console, *I love programming so much!*

Learning, Logic, Optimization

9: Create a function that finds the missing leg length of a triangle with a known hypotenuse and leg length, and use it to find the leg length of a triangle with side length 68 and hypotenuse 293.

Logic, Math

10: In less than 3 lines, use a (for) loop to print the numbers 0 to 9 on different lines in the console. (Note: loops will be coming up a lot in future problems. Learn them well!)

Learning, Logic, Optimization

11: The sum of the first 5 natural numbers is 15, as $15 = 1 + 2 + 3 + 4 + 5$. Find the sum of the first 1000 natural numbers.

Logic

12: Create a function that returns whether an inputted number is odd or even in terms of false or true respectively. 5 should return false, and 10820398928 should return true. Then, evaluate if the square root of 400 divided by 10 is even or not.

Learning, Logic

13: If we list all natural numbers below 10 that are multiples of 3 and 5, we get 3, 5, 6, and 9. The sum of these multiples is 23. Find the sum of all of the multiples of 3 or 5 below 1000.

Logic, Euler (1, <https://projecteuler.net/problem=1>)

14: Find the sum of the two 20-digit numbers 38729813798127398732 and 21927398012792939242.

Learning

15: Create a list containing all of the prime numbers from 2 to 2000. Find the length of this list with a method.

16: Sort the following 100 numbers from least to greatest.

{84, 41, 92, 23, 97, 11, 1, 6, 64, 63, 30, 72, 69, 80, 10, 69, 59, 17, 70, 9, 5, 20, 31, 83, 84, 94, 42, 39, 69, 29, 73, 40, 70, 39, 50, 33, 71, 70, 85, 59, 53, 72, 54, 62, 22, 42, 11, 69, 57, 87, 91, 9, 8, 85, 18, 2, 1, 71, 95, 45, 87, 77, 44, 55, 53, 63, 83, 74, 54, 47, 86, 61, 66, 56, 42, 51, 33, 29, 39, 83, 6, 84, 36, 54, 100, 7, 37, 15, 45, 81, 93, 50, 39, 51, 57, 59, 68, 82, 65, 95}

17: In this 100 number list, replace all "1"s with "a", all "2"s with "b", so on and so forth.

{11, 2, 16, 19, 4, 26, 6, 6, 3, 19, 19, 20, 21, 15, 12, 24, 10, 3, 13, 13, 21, 12, 6, 10, 12, 20, 17, 22, 15, 26, 12, 13, 21, 22, 10, 14, 1, 14, 13, 11, 26, 26, 17, 18, 9, 22, 1, 23, 7, 16, 14, 21, 8, 3, 6, 24, 6, 6, 11, 1, 5, 13, 8, 1, 15, 15, 24, 15, 9, 23, 21, 6, 24, 22, 13, 9, 18, 22, 22, 15, 4, 25, 14, 3, 23, 7, 3, 12, 5, 11, 17, 7, 9, 3, 16, 25, 5, 23, 8, 24}