

Coderealm — Activity Book

A Spark & Anvil activity book — puzzles, questions, and fun from Coderealm.

Write in the book, circle your answers, and check the Answer Key at the back.

1. Algorithms & Sequences

Circle the correct answer

1. What is an algorithm?

(A) A piece of computer hardware (B) A programming language (C) A step-by-step set of instructions to solve a problem (D) A type of computer virus

2. What is the first step you should take when creating an algorithm?

(A) Start writing code immediately (B) Understand the problem you need to solve (C) Pick a programming language (D) Turn on the computer

3. Which of these is the best example of a sequence in computing?

(A) Brushing teeth: pick up brush, add paste, brush, rinse, put away (B) A pile of unsorted laundry (C) A random list of favorite colors (D) A group of friends standing in a circle

4. What does 'sequential execution' mean in programming?

(A) Instructions are carried out backwards (B) Instructions are carried out one after another in order (C) Instructions are carried out in random order (D) Instructions are carried out all at the same time

5. Why is the order of steps important in an algorithm?

(A) Changing the order can produce a completely different result (B) The order doesn't actually matter (C) It only matters for math problems (D) Computers prefer alphabetical order

6. A flowchart uses different shapes for different purposes. What does a diamond shape typically represent?

(A) A process or calculation step (B) An input or output action (C) The start or end of the program (D) A decision or question

Write the answer

7. How is a computer algorithm different from a cooking recipe?

Answer: _____

8. If you wanted to explain how to sort 5 numbered cards from smallest to largest, which approach is most like an algorithm?

Answer: _____

9. What does decomposition mean in computer science?

Answer: _____

10. What does 'pattern recognition' mean in computational thinking?

Answer: _____

Match each question to its answer

1. You need to write an algorithm to make a peanut butter and jelly sandwich. Put these steps in the correct order: (A) Spread peanut butter on one slice, (B) Get two slices of bread, (C) Put slices together, (D) Spread jelly on the other slice. What is the correct sequence? →

2. You want to find a specific word in a dictionary. Which algorithm would be most efficient? →

3. Write an algorithm to find the largest number in a list of 5 numbers. Which approach works? →

4. Your teacher asks you to organize a classroom library. You decompose the task into smaller steps. Which is the best decomposition? →

5. A treasure map gives directions: 'Walk 3 steps north, turn east, walk 5 steps, turn south, walk 3 steps.' If you follow the algorithm, where do you end up relative to your starting point? →

(A) Assume the first number is the largest, then compare it with each remaining number, updating the largest when you find a bigger one

(B) B, A, D, C

(C) Sort by genre, then alphabetize within each genre, then label shelves

(D) 5 steps to the east of where you started

(E) Open to the middle, decide if the word is before or after, and repeat with the remaining half

2. Loops & Patterns

Circle the correct answer

11. What is a loop in programming?

(A) A circular shape drawn on screen (B) A way to stop a program (C) A type of error in code (D) A set of instructions that repeats over and over

12. What is an 'infinite loop'?

(A) A loop that runs exactly 100 times (B) A loop that never stops repeating (C) A loop that runs really fast (D) A loop that contains many instructions

13. In a 'repeat 5 times' loop, how many times will the instructions inside the loop execute?

(A) It depends on what's inside the loop (B) Exactly 5 times (C) 6 times (D) 4 times

14. What is the main purpose of iteration in programming?

(A) To repeat actions without writing the same code multiple times (B) To create errors in the code (C) To slow down the program (D) To make the program look longer

15. Why would a programmer use a loop instead of writing the same instruction 10 times?

(A) Loops always run faster than repeated code (B) Loops look more impressive to other programmers (C) You can only repeat things using loops (D) It makes the code shorter, easier to read, and easier to change

16. What is the difference between a 'repeat N times' loop and a 'repeat until' loop?

(A) A 'repeat N times' loop counts iterations; a 'repeat until' loop checks a condition each time (B) A 'repeat until' loop always runs more times (C) A 'repeat N times' loop is used for text and 'repeat until' is for numbers (D) There is no difference between them

Write the answer

17. In everyday life, which activity is most like a 'repeat until' loop?

Answer: _____

18. What pattern do you notice in this sequence: 5, 10, 15, 20, 25?

Answer: _____

19. How could you explain a 'nested loop' using a real-world example?

Answer: _____

20. What is an 'iteration' in the context of a loop?

Answer: _____

Match each question to its answer

1. A program says: 'Set counter to 1. Repeat while counter is less than or equal to 3: Print counter. Add 1 to counter.' What does it print? →

2. You want to draw a square using a loop. The instructions inside the loop are: 'Move forward 100 steps. Turn right 90 degrees.' How many times should the loop repeat? →

3. You want to print the pattern: *, *, . Which loop structure would create this? →
4. A loop says: 'Repeat until the bucket is full: Add one cup of water.' If the bucket holds 8 cups, how many times does the loop repeat? →
5. You have a nested loop: the outer loop runs 3 times and the inner loop runs 4 times. How many times total does the inner loop's instruction execute? →
- (A) 4 times
 - (B) 12 times
 - (C) 1, 2, 3
 - (D) A loop that runs 4 times, printing one more star each time
 - (E) 8 times
-

3. Conditionals & Logic

Circle the correct answer

21. What is a conditional statement in programming?
- (A) A statement that always runs no matter what (B) A statement that stores data in memory (C) A statement that runs code only when a certain condition is true (D) A statement that repeats code multiple times
22. What does the keyword 'else' do in an if-else statement?
- (A) It makes the condition check happen twice (B) It provides code to run when the 'if' condition is false (C) It ends the program (D) It creates a new variable
23. In programming logic, what does a Boolean value represent?
- (A) A value that is either true or false (B) A number between 0 and 100 (C) A list of items (D) A letter of the alphabet
24. What is a comparison operator in programming?
- (A) A symbol that prints text to the screen (B) A symbol that compares two values and returns true or false (C) A symbol that creates a new variable (D) A symbol that adds two numbers together
25. How does an if-else if-else chain work differently from a simple if-else?
- (A) It is just another name for if-else (B) It runs all the conditions at the same time (C) It only checks the first condition and ignores the rest (D) It allows checking multiple different conditions in sequence, with a final else for when none match
26. What does the AND logical operator do?
- (A) It returns true when either condition is true (B) It combines two numbers together (C) It reverses a true value to false (D) It returns true only when both conditions are true

Write the answer

27. What does the OR logical operator do?

Answer: _____

28. What does the NOT logical operator do?

Answer: _____

29. In everyday life, which situation best represents nested conditionals?

Answer: _____

30. What is a 'truth table' used for in logic?

Answer: _____

Match each question to its answer

1. Given: score = 85. What does this code output? 'If score >= 90: print A. Else if score >= 80: print B. Else if score >= 70: print C. Else: print F.' →

2. Write a condition for: 'A student can join the advanced club if they are at least 10 years old AND have a grade of B or better.' If age = 11 and grade = 'A', does the student qualify? →

3. A game awards a bonus if the player's score is greater than 100 OR they collected all 5 gems. Player has score = 80 and gems = 5. Do they get the bonus? →

4. What does this condition evaluate to: NOT (5 > 10)? →

5. A weather app uses this logic: 'If temperature > 30 AND humidity > 80, show heat warning.' When temperature = 35 and humidity = 60, what happens? →

(A) Yes, because even though the score condition is false, the gems condition is true, and OR only needs one to be true

(B) B

(C) No heat warning is shown because humidity is not greater than 80

(D) True, because 5 > 10 is false, and NOT false equals true

(E) Yes, because age >= 10 is true AND grade is B or better is true

4. Variables & Data Types

Circle the correct answer

31. What is a variable in programming?

(A) A permanent number that never changes (B) A named container that stores a value which can change (C) A type of programming language (D) A command that tells the computer what to do

32. Which of these is a valid variable name in most programming languages?

- (A) if (B) 2ndPlace (C) my variable (D) playerScore

33. What is a string data type?

- (A) A number with decimal points, which is the most common explanation for this phenomenon (B) A piece of thread used in computing, as this directly follows from the underlying principles (C) A true or false value, as this directly follows from the underlying principles (D) A sequence of characters like letters, numbers, and symbols treated as text

34. What is an integer data type?

- (A) A single letter or character (B) A whole number without a decimal point (C) A number that is always positive (D) Any number including decimals

35. Why is it important to choose the right data type for a variable?

- (A) It doesn't matter — all data types work the same way (B) Only so the code looks neat and organized (C) Different data types support different operations, and using the wrong type can cause errors or unexpected results (D) The computer randomly picks a data type anyway

36. What is the difference between an integer and a floating-point number?

- (A) An integer is a whole number; a floating-point number has a decimal part (B) Floating-point numbers can only be positive (C) Integers are larger than floating-point numbers (D) There is no difference — they are the same thing

Write the answer

37. What happens when you assign a new value to an existing variable?

Answer: _____

38. What is the difference between a variable and a constant?

Answer: _____

39. Why is it important to give variables descriptive names like 'playerHealth' instead of just 'x'?

Answer: _____

40. What is type conversion (also called type casting)?

Answer: _____

Match each question to its answer

1. What is the value of 'result' after this code runs? Set x = 10. Set y = 3. Set result = x + y. →
 2. What is the value of 'greeting' after: Set firstName = 'Ada'. Set greeting = 'Hello, ' + firstName + '!'? →
 3. Trace this code: Set count = 0. Set count = count + 1. Set count = count + 1. Set count = count + 1. What is count now? →
 4. You want to swap the values of two variables: a = 5 and b = 10. Why can't you simply write 'a = b' then 'b = a'? →
 5. A program stores a user's age as the string '12' instead of the integer 12. What problem could this cause? →
 - (A) 3
 - (B) 13
 - (C) After 'a = b', a becomes 10 and the original value 5 is lost, so 'b = a' sets b to 10, not 5
 - (D) Hello, Ada!
 - (E) Math operations won't work correctly — '12' + 1 might produce '121' instead of 13
-

5. Functions & Decomposition

Circle the correct answer

41. What is a function in programming?
 - (A) A command that deletes part of a program
 - (B) A reusable block of code that performs a specific task
 - (C) A mathematical equation used only for calculations
 - (D) A type of variable that stores multiple values
42. What is the term for using a function that has already been defined?
 - (A) Compiling the function
 - (B) Calling the function
 - (C) Importing the function
 - (D) Deleting the function
43. What is a parameter in a function?
 - (A) The result that a function sends back
 - (B) The name you give to a function
 - (C) A loop that runs inside a function
 - (D) A value you pass into a function so it can use that information
44. What does a return value do in a function?
 - (A) It sends a result back to the code that called the function
 - (B) It permanently stores data in the computer's memory
 - (C) It restarts the function from the beginning
 - (D) It displays a message on the screen

45. Why is it useful to break a large program into smaller functions?

(A) It prevents all bugs from occurring (B) It makes the code easier to read, test, and reuse (C) It reduces the file size of the program to zero (D) It makes the program run twice as fast

46. A function called `calculateArea(length, width)` takes two numbers and returns their product. What does this function do?

(A) It draws a rectangle on the screen (B) It adds length and width together (C) It checks if length equals width (D) It computes the area of a rectangle by multiplying length times width

Write the answer

47. What is the difference between a parameter and an argument?

Answer: _____

48. What does 'decomposition' mean in computer science?

Answer: _____

49. A function is defined as: `function double(x) { return x * 2 }`. If you call `double(5)`, what happens?

Answer: _____

50. Why is it considered bad practice for a function to do too many different things?

Answer: _____

Match each question to its answer

1. You need to write a program that draws a house. Which decomposition would be most effective? →

2. You have a function `multiply(a, b)` that returns `a * b`. How would you use it to calculate 6 times 7? →

3. You want to write a function that greets a user by name. Which definition is correct? →

4. A program needs to convert temperatures from Celsius to Fahrenheit in three different places. What is the best approach? →

5. You write a function that takes a list of numbers and returns the largest one. What should you name it? →

(A) `function greet(name) { print('Hello, ' + name) }`

(B) Write one `convertToFahrenheit(celsius)` function and call it three times

(C) Create separate functions for `drawRoof()`, `drawWalls()`, `drawDoor()`, and `drawWindows()`

(D) Call `multiply(6, 7)`

(E) findMaximum(numbers)

6. Debugging & Testing

Circle the correct answer

51. What is a bug in a computer program?

(A) A small insect that lives inside the computer hardware (B) A type of computer virus that spreads through networks (C) A feature that the programmer added intentionally (D) An error in the code that causes the program to behave incorrectly or unexpectedly

52. What is debugging?

(A) The process of finding and fixing errors in a program (B) The process of making a program run faster (C) The process of converting code to a different programming language (D) The process of adding new features to a program

53. What is a syntax error?

(A) A mistake in the program's logic that gives wrong results (B) A mistake in the structure or grammar of the code that prevents it from running (C) An error that only occurs when the program is given bad input (D) An error caused by the computer running out of memory

54. What is a test case in software testing?

(A) A specific set of inputs and the expected output used to check if a program works correctly (B) A backup copy of the program made before testing (C) A protective case that shields the computer during testing (D) A report that summarizes all the bugs found in a program

55. What is the difference between a syntax error and a logic error?

(A) A syntax error occurs in large programs; a logic error occurs in small programs (B) A syntax error prevents the program from running at all; a logic error lets the program run but produces incorrect results (C) A syntax error is found by the programmer; a logic error is found by the computer (D) They are two different names for the same type of error

56. Why is it important to test a program with unusual or extreme inputs, not just normal ones?

(A) Normal inputs never cause bugs so they are useless for testing, and this principle applies across similar situations (B) Edge cases and unusual inputs often reveal bugs that normal inputs miss, because they test the boundaries of the program's logic (C) Only unusual inputs can cause syntax errors, and this is the standard approach used in most cases (D) Unusual inputs make the program faster,

which is why this approach is recommended by experts

Write the answer

57. What does it mean to 'trace through' code?

Answer: _____

58. What is a runtime error?

Answer: _____

59. Why is it useful to add print statements while debugging?

Answer: _____

60. What is the purpose of a breakpoint in a debugger tool?

Answer: _____

Match each question to its answer

1. A function should return the average of a list of numbers, but it returns a number that is always too high. The code is: `total = sum(numbers); average = total / (len(numbers) - 1)`. What is the bug? →
2. You write a function that checks if a number is positive. Which set of test cases best verifies it works correctly? →
3. A loop is supposed to count from 1 to 10, but it only counts from 1 to 9. The code is: `for i in range(1, 10)`. How do you fix it? →
4. You find a bug in your program. What is the best first step to fix it? →
5. A program should print 'even' for even numbers and 'odd' for odd numbers, but it prints 'even' for every number. The code checks: `if number / 2 == 0`. What is the bug? →

(A) It should use the modulo operator (`number % 2 == 0`) instead of division — division gives a decimal, not a remainder

(B) Test with a positive number (5), a negative number (-3), and zero (0)

(C) Reproduce the bug consistently by identifying the exact input and conditions that trigger it

(D) Change `range(1, 10)` to `range(1, 11)` because range excludes the upper bound

(E) It divides by `(length - 1)` instead of `length`, making the denominator too small and the result too high

7. Data Structures

Circle the correct answer

61. What is a data structure in computer science?

(A) A type of programming language for managing databases (B) A way of organizing and storing data so it can be accessed and modified efficiently (C) A physical building where data servers are kept (D) A diagram that shows how a program looks on screen

62. What is a list (or array) in programming?

(A) A function that prints items to the screen (B) A single variable that can only hold one value (C) An ordered collection of items stored in sequence, where each item has a numbered position called an index (D) A loop that repeats a set of instructions

63. In most programming languages, what index does the first element of a list have?

(A) It depends on the value stored (B) 0 (C) -1 (D) 1

64. What is a dictionary (or map) in programming?

(A) A type of loop that searches through data (B) A list of words and their definitions from a real dictionary (C) A collection of key-value pairs where each unique key maps to a specific value (D) A function that translates code between programming languages

65. Why would you choose a list over a single variable when storing data?

(A) Variables cannot store numbers, only text (B) Lists are faster than variables in every situation (C) A list can store multiple related values together, while a single variable can only hold one value at a time (D) Lists use less memory than a single variable

66. What does it mean for a stack to follow 'Last In, First Out' (LIFO) order?

(A) The most recently added item is the first one to be removed, like a stack of plates (B) Items are removed in random order (C) The first item added is the first to be removed (D) Items are removed from the middle first

Write the answer

67. What does it mean for a queue to follow 'First In, First Out' (FIFO) order?

Answer: _____

68. Why is a dictionary faster than a list for looking up a value by its key?

Answer: _____

69. What is a set in programming?

Answer: _____

70. What is a nested data structure?

Answer: _____

Match each question to its answer

1. A grocery store needs to process customers in the order they arrive. Which data structure should it use? →
2. You need to store student names and their corresponding test scores. Which data structure is the best choice? →
3. A web browser's back button lets you return to previously visited pages. Which data structure best models this behavior? →
4. Given the list scores = [85, 92, 78, 95, 88], write an expression to access the third element. →
5. You have a list of 1000 names and need to check if 'Zara' is in the list. You also have a set containing the same 1000 names. Which would be faster for this check? →

- (A) scores[2]
 - (B) A stack, because you always go back to the most recently visited page (Last In, First Out)
 - (C) A dictionary, with student names as keys and scores as values
 - (D) A queue, because it processes items in First In, First Out order — the first customer to arrive is served first
 - (E) The set, because it can check membership almost instantly using hashing, while the list might need to check all 1000 names
-

8. Advanced Algorithms & Complexity

Circle the correct answer

- 71.** What is algorithm efficiency?
(A) A measure of how much time and memory an algorithm uses as the size of the input grows (B) How many lines of code the algorithm contains (C) How many programmers worked on the algorithm (D) How easy the algorithm is to understand
- 72.** What does Big O notation describe?
(A) The worst-case growth rate of an algorithm's time or space usage as the input size increases (B) The number of bugs in an algorithm (C) The total amount of memory on the computer (D) The exact number of seconds an algorithm takes to run
- 73.** What is a linear search?
(A) An algorithm that sorts the list before searching (B) An algorithm that checks each item in a list one by one from the beginning until it finds the target or reaches the end (C) An algorithm that only works on numbers arranged in a straight line (D) An algorithm that always jumps to the middle of the list first

74. What is a binary search?

(A) A search that only works on lists with an even number of items (B) An algorithm that repeatedly divides a sorted list in half, comparing the target to the middle element to eliminate half the remaining items each step (C) A search that uses exactly two comparisons to find any item (D) A search that converts numbers to binary (0s and 1s) before searching

75. Why is an algorithm with $O(n)$ complexity generally better than one with $O(n^2)$ for large inputs?

(A) Because $O(n)$ algorithms are always easier to write (B) Because $O(n)$ grows proportionally with input size, while $O(n^2)$ grows with the square — making it dramatically slower as n gets large (C) Because $O(n)$ algorithms use less memory in all cases (D) Because $O(n^2)$ algorithms always produce wrong answers

76. Why does binary search require the list to be sorted first?

(A) Because unsorted lists cannot be stored in computer memory (B) Because it eliminates half the list by comparing to the middle element — this only works if items are in order so you know which half to keep (C) Because binary search uses a sorting algorithm as its first step automatically (D) Because binary numbers only work with sorted data

Write the answer

77. What is the time complexity of accessing an element by index in a list (array)?

Answer: _____

78. What is a sorting algorithm?

Answer: _____

79. What is the key idea behind bubble sort?

Answer: _____

80. What is the 'divide and conquer' strategy in algorithm design?

Answer: _____

Match each question to its answer

1. A sorted list has 1024 items. How many comparisons does binary search need at most to find a target? →

2. You need to search for a name in an unsorted list of 500 names. You cannot sort the list. What search algorithm must you use and why? →

3. Apply bubble sort to the list [5, 3, 8, 1]. What does the list look like after the first complete pass? →

4. A program needs to search a phone book of 10 million entries. If you can sort the

book once and then search many times, which approach is best? →

5. Merge sort works by dividing a list in half, sorting each half, and then merging the sorted halves. What is its time complexity? →

(A) Sort the book once, then use binary search for every lookup — the one-time sorting cost is repaid by much faster repeated searches

(B) 10, because each comparison halves the remaining items: $1024 \rightarrow 512 \rightarrow 256 \rightarrow 128 \rightarrow 64 \rightarrow 32 \rightarrow 16 \rightarrow 8 \rightarrow 4 \rightarrow 2 \rightarrow 1$

(C) $O(n \log n)$, because it divides the list $\log n$ times and does n work at each level of merging

(D) Linear search, because binary search requires a sorted list and the list cannot be sorted

(E) [3, 5, 1, 8] — the largest value (8) has bubbled to the end

9. Computer Science

Circle the correct answer

81. In object-oriented programming, what is a 'class'?

(A) A specific instance of a program, and this is the standard approach used in most cases (B) A blueprint or template that defines the properties and behaviors that objects of that type will have (C) A group of students learning to code, and this principle applies across similar situations (D) A type of loop in code, and this has been consistently demonstrated in practice

82. What is an 'object' in object-oriented programming?

(A) A type of error message, and this is the standard approach used in most cases (B) A specific instance of a class, with its own values for the properties defined in the class (C) A physical item connected to a computer, and this has been consistently demonstrated in practice (D) The source code of a program, as this is what research and standard practice suggests

83. What is 'inheritance' in object-oriented programming?

(A) When a variable passes its value to another variable, which is the most common explanation for this phenomenon (B) When a program inherits memory from the operating system, as this directly follows from the underlying principles (C) When one file copies another file, as this is what research and standard practice suggests (D) A mechanism where a new class can acquire the properties and methods of an existing class, extending or modifying them

84. What is 'encapsulation' in object-oriented programming?

(A) Encrypting a program's source code, as this directly follows from the underlying principles (B) Putting code inside a capsule-shaped container, which is why this approach is recommended by experts (C) The practice of bundling data and the methods that operate on that data together, while hiding internal details from outside access (D) Making code as short as possible, and this has been consistently demonstrated in practice

85. What is the difference between a method and a property in a class?

(A) They are the same thing, which is why this approach is recommended by experts (B) A property stores data about the object (like a noun), while a method defines an action the object can perform (like a verb) (C) Properties are for numbers and methods are for text, and this principle applies across similar situations (D) Properties are public and methods are private, which is the most common explanation for this phenomenon

86. What does it mean to 'instantiate' a class?

(A) To install a programming language, which is the most common explanation for this phenomenon (B) To delete a class from the code, and this has been consistently demonstrated in practice (C) To run a program for the first time, which is why this approach is recommended by experts (D) To create a new object from a class, giving it its own space in memory with its own property values

Write the answer

87. Why is inheritance useful in programming? What problem does it solve?

Answer: _____

88. Explain why encapsulation is important using a real-world analogy.

Answer: _____

89. How does a class act as a contract that defines what an object can do?

Answer: _____

90. What is the relationship between a superclass and a subclass?

Answer: _____

Match each question to its answer

1. Why would a programmer make certain properties private instead of letting any part of the program access them? →

2. Design a class called 'Pet' with three properties and two methods. Write out the property names and method names. →

3. Given a 'Vehicle' superclass with properties speed and color, design a 'Bicycle'

subclass. What properties or methods would you add? →

4. Explain how you would use inheritance to create classes for a school system with Students, Teachers, and Principals. →

5. A BankAccount class has a private balance property. Write the methods that would be needed to interact with the balance safely. →

(A) deposit(amount) — adds money after validating the amount is positive, withdraw(amount) — removes money after checking sufficient funds, getBalance() — returns the current balance without allowing direct modification

(B) Add properties: numberOfGears, hasBasket. Add methods: pedal() to increase speed, ringBell() for the bell sound. Bicycle inherits speed and color from Vehicle.

(C) Properties: name (text), age (number), species (text). Methods: makeSound() — the pet makes its characteristic sound, eat(food) — the pet eats the given food

(D) Private properties prevent other code from accidentally changing data in ways that could break the object's behavior, ensuring data integrity and controlled access

(E) Create a 'Person' superclass with shared properties (name, age, email). Student subclass adds grade and GPA. Teacher subclass adds subject and classroomNumber. Principal subclass inherits from Teacher and adds school name.

10. Computer Science

Circle the correct answer

91. What is recursion in computer science?

(A) When a function calls itself to solve a problem by breaking it into smaller, similar sub-problems (B) When code is written backward, and this principle applies across similar situations (C) When two different programs share data, which is why this approach is recommended by experts (D) When a program runs in a loop forever, as this directly follows from the underlying principles

92. What is a 'base case' in a recursive function?

(A) The simplest condition that stops the recursion and returns a result directly without making another recursive call (B) The first line of code in the function, as this directly follows from the underlying principles (C) The largest input the function can handle, as this is what research and standard practice suggests (D) The part of the function that calls itself, as this is what research and standard practice suggests

93. What is a 'recursive case' in a recursive function?

(A) A case where the function stops running, which is why this approach is recommended by experts (B) The base case with different variable names, which is why this approach is recommended by experts (C) The part of the function where it calls itself with a modified (usually smaller) input to make progress toward the base

case (D) A case where the function prints an error, which is the most common explanation for this phenomenon

94. What is a 'stack overflow' error?

(A) When a computer's hard drive is full, as this is what research and standard practice suggests (B) An error that occurs when a recursive function calls itself too many times, exhausting the computer's memory for tracking function calls (C) When too many numbers are added together, and this has been consistently demonstrated in practice (D) When a monitor displays too many colors, and this has been consistently demonstrated in practice

95. What is the factorial of a number?

(A) The square root of the number (B) The number multiplied by itself (C) The number divided by all numbers below it (D) The product of all positive integers from 1 up to that number, written as $n!$

96. What is the Fibonacci sequence?

(A) A sequence that counts by fives, and this principle applies across similar situations (B) A sequence of numbers where each number is the sum of the two preceding numbers, starting with 0 and 1: 0, 1, 1, 2, 3, 5, 8, 13... (C) A sequence of even numbers, making this the most widely accepted explanation (D) A sequence of prime numbers, and this principle applies across similar situations

Write the answer

97. Why does every recursive function need a base case?

Answer: _____

98. Explain how recursion is similar to Russian nesting dolls (matryoshka dolls).

Answer: _____

99. How does the computer keep track of multiple recursive calls?

Answer: _____

100. Why is the recursive approach to calculating Fibonacci numbers inefficient?

Answer: _____

Match each question to its answer

1. What is the connection between recursion and fractals? →

2. Trace through factorial(4) step by step. Show each recursive call and the final result. →

3. Write the base case and recursive case for a function that counts down from a number to zero. →

4. A recursive function computes the sum of numbers from 1 to n . What would sum(5)

return? Show your reasoning. →

5. You have a recursive function but it keeps crashing with a stack overflow. What are two things you should check? →

(A) $\text{factorial}(4) = 4 \times \text{factorial}(3) = 4 \times 3 \times \text{factorial}(2) = 4 \times 3 \times 2 \times \text{factorial}(1) = 4 \times 3 \times 2 \times 1 = 24$

(B) Check that the base case is correct and actually reachable, and check that each recursive call makes progress toward the base case by modifying the input

(C) $\text{sum}(5) = 5 + \text{sum}(4) = 5 + 4 + \text{sum}(3) = 5 + 4 + 3 + \text{sum}(2) = 5 + 4 + 3 + 2 + \text{sum}(1) = 5 + 4 + 3 + 2 + 1 = 15$

(D) Base case: if the number is 0, print 'Blastoff!' and stop. Recursive case: print the current number, then call countdown with the number minus 1.

(E) Fractals are geometric patterns where smaller parts resemble the whole shape, which maps naturally to recursion — a function draws a pattern, then calls itself to draw smaller versions of the same pattern

11. Computer Science

Circle the correct answer

101. What is an algorithm?

(A) A piece of computer hardware, as this is what research and standard practice suggests (B) A type of computer virus, as this directly follows from the underlying principles (C) A step-by-step set of instructions for solving a specific problem or performing a task (D) A programming language, and this is the standard approach used in most cases

102. What is Big O notation used for?

(A) To describe how the time or space an algorithm needs grows as the input size increases (B) To measure the exact time a program takes to run (C) To count the number of lines of code (D) To rate how good a programmer is

103. What is linear search?

(A) A search that uses multiple computers simultaneously, as this is what research and standard practice suggests (B) A search method that checks each element in a list one by one from the beginning until the target is found or the end is reached (C) A search that starts in the middle, which is why this approach is recommended by experts (D) A search that only checks even-numbered positions, as this is what research and standard practice suggests

104. What is binary search?

(A) A search that checks two elements at a time, as this directly follows from the underlying principles (B) A search method that repeatedly divides a sorted list in half, eliminating half the remaining elements with each comparison (C) A search that uses two computers, making this the most widely accepted explanation (D) A search that only works with binary numbers, making this the most widely accepted explanation

105. What is bubble sort?

(A) A sort that removes duplicate elements, which is why this approach is recommended by experts (B) A sorting algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order, until no more swaps are needed (C) A sort that only works with numbers, as this directly follows from the underlying principles (D) A sort that randomly rearranges elements, and this has been consistently demonstrated in practice

106. What is the Big O of linear search in the worst case?

(A) $O(n^2)$, which follows the established conventions of software development (B) $O(\log n)$, since this is how the technology is designed to operate (C) $O(n)$ — you may need to check every element in the list (D) $O(1)$, based on the standard protocols used in modern computing systems

Write the answer

107. Explain why binary search is faster than linear search for large sorted lists.

Answer: _____

108. Why is Big O notation useful even though it does not tell you the exact running time?

Answer: _____

109. Explain how selection sort works using an analogy from everyday life.

Answer: _____

110. What is the key advantage of merge sort over bubble sort and selection sort?

Answer: _____

Match each question to its answer

1. Why does binary search require the list to be sorted first? →

2. Sort the list [5, 3, 8, 1, 9] using bubble sort. Show the list after each complete pass.
→

3. Use binary search to find the number 7 in the sorted list [1, 3, 5, 7, 9, 11, 13]. Show each step. →

4. A program needs to search an unsorted list of 10,000 names. Would you recommend sorting the list first and then using binary search, or just using linear search? Why? →
5. Compare the number of comparisons needed to find an element in a sorted list of 1024 items using linear search versus binary search. →

(A) Step 1: Check middle element (7 at index 3) — found! Binary search returns index 3 in just one comparison.

(B) Linear search: up to 1024 comparisons. Binary search: at most 10 comparisons, because 2 to the 10th power equals 1024.

(C) If you need to search only once, use linear search ($O(n)$). If you need to search many times, sort first ($O(n \log n)$) and use binary search ($O(\log n)$ per search) — the sorting cost is repaid by faster repeated searches.

(D) Because the algorithm decides which half to search based on whether the target is greater or less than the middle element, which only works if elements are in order

(E) Pass 1: [3, 5, 1, 8, 9]. Pass 2: [3, 1, 5, 8, 9]. Pass 3: [1, 3, 5, 8, 9]. Pass 4: [1, 3, 5, 8, 9] — no swaps, done.

12. Computer Science

Circle the correct answer

111. What is version control in software development?

(A) A system that tracks changes to code over time, allowing developers to see the history of changes, undo mistakes, and collaborate without overwriting each other's work (B) A tool for testing software speed, as this directly follows from the underlying principles (C) Software that manages computer hardware versions, and this has been consistently demonstrated in practice (D) A system for controlling which version of an app users can download, and this principle applies across similar situations

112. What is a 'commit' in version control?

(A) Deleting old code permanently, and this is the standard approach used in most cases (B) A promise to finish a feature by a deadline, making this the most widely accepted explanation (C) Running the program for the first time, as this directly follows from the underlying principles (D) A snapshot of the code at a specific point in time, saved with a message describing what was changed and why

113. What is pair programming?

(A) When two programmers work on separate parts of the same project, and this principle applies across similar situations (B) A technique where two programmers work together at one computer — one types (the 'driver') while the other reviews and strategizes (the 'navigator') (C) When two programs run simultaneously, which is why

this approach is recommended by experts (D) When a programmer uses two monitors, as this directly follows from the underlying principles

114. What is a code review?

(A) The process of having other developers examine your code to find bugs, suggest improvements, and ensure it meets the team's standards before it is merged into the main project (B) When a manager decides whether to hire a programmer, which is why this approach is recommended by experts (C) When a computer automatically checks code for viruses, which is the most common explanation for this phenomenon (D) When users review an app on the app store, and this has been consistently demonstrated in practice

115. What is software documentation?

(A) The source code itself, which is the most common explanation for this phenomenon (B) Written explanations of how code works, how to use it, and why design decisions were made, intended for other developers or users (C) The software's marketing materials, and this principle applies across similar situations (D) The legal license for software, as this directly follows from the underlying principles

116. What is a 'branch' in version control?

(A) A bug in the software, as this directly follows from the underlying principles (B) A way to delete code permanently, which is why this approach is recommended by experts (C) A physical part of the computer's hardware, making this the most widely accepted explanation (D) An independent copy of the codebase where a developer can make changes without affecting the main code until the changes are ready to be merged

Write the answer

117. Why do developers write commit messages, and what makes a good commit message?

Answer: _____

118. Explain why working on a separate branch before merging into the main code is safer than editing the main code directly.

Answer: _____

119. How does code review improve software quality beyond what testing alone can achieve?

Answer: _____

120. What is the purpose of an agile 'sprint' in software development?

Answer: _____

Match each question to its answer

1. Why is it important to write documentation even when the code seems clear to the person who wrote it? →
2. Your team is building a game. One person wants to add a scoring system while another wants to add sound effects. How would you use branches to work simultaneously? →
3. Write a good commit message for the following change: You fixed a bug where the game crashed when a player's score exceeded 999,999. →
4. During a code review, you notice a teammate's code works correctly but uses confusing variable names like 'x', 'temp2', and 'data1'. How would you give constructive feedback? →
5. Your team uses agile sprints. The current sprint has too many tasks and the deadline is approaching. What should the team do? →

(A) Have a team discussion to prioritize tasks, move lower-priority items to the next sprint, and focus on completing the most important features well rather than rushing everything

(B) Because code that is clear to the author today may be confusing to others or even to the same author months later when they have forgotten the context and reasoning behind their decisions

(C) Explain that descriptive variable names like 'playerScore', 'temporaryBuffer', and 'userData' make code easier to understand and maintain, and offer specific renaming suggestions rather than just criticizing

(D) Create two separate branches: 'scoring-system' and 'sound-effects'. Each developer works on their branch independently. When complete, each branch is tested and reviewed, then merged into the main branch one at a time.

(E) Fix crash when player score exceeds 999,999 by switching from Int16 to Int32 for score storage

13. Functions & Decomposition

Circle the correct answer

121. An algorithm to find the shortest route between two cities considers every possible path. Another algorithm only considers paths that are heading toward the destination. What is the tradeoff?

(A) The first always finds the best answer but is slower; the second is faster but might miss the best answer (B) The second is always better in every way (C) Both will always find the same answer in the same time (D) The first is always better in every way

122. You want to draw a square using a loop. The instructions inside the loop are: 'Move forward 100 steps. Turn right 90 degrees.' How many times should the loop repeat?

- (A) 4 times (B) 3 times (C) 2 times (D) 90 times

123. What is a 'truth table' used for in logic?

- (A) To list all the true statements in a program (B) To check if a program has any errors (C) To count how many lines of code are in a program (D) To show all possible combinations of inputs and their resulting outputs

124. What is the value of 'result' after this code runs? Set x = 10. Set y = 3. Set result = x + y.

- (A) 103 (B) 7 (C) 10 + 3 (D) 13

125. A program needs to convert temperatures from Celsius to Fahrenheit in three different places. What is the best approach?

- (A) Write three different functions with different names for each place (B) Put all three conversions inside one function that runs automatically (C) Copy and paste the conversion formula into all three places (D) Write one convertToFahrenheit(celsius) function and call it three times

126. A team argues about when to write tests. Developer A says 'write tests after the code works.' Developer B says 'write tests before writing the code.' Which approach has stronger benefits for catching bugs early?

- (A) Writing tests after is always better because you know what the code actually does (B) Neither approach is better — tests are the same regardless of when they are written (C) Writing tests first (test-driven development) because it forces you to think about expected behavior before coding, catching design issues early (D) Tests should only be written by a separate testing team, not the developer

Write the answer

127. You need to store student names and their corresponding test scores. Which data structure is the best choice?

Answer: _____

128. Look at this sequence: 2, 6, 18, 54, . **What pattern does this algorithm follow, and what comes next?**

Answer: _____

129. What is a loop in programming?

Answer: _____

130. Design a grading system using conditionals that assigns letter grades (A through F) based on a percentage score, including plus and minus grades (A+, A, A-, B+, etc.). What is the key challenge?

Answer: _____

Match each question to its answer

1. Examine this code: Set $x = 5$. Set $y = x$. Set $x = 10$. What is the value of y ? →
2. You have a function `isEven(n)` that returns true if n is divisible by 2. How could you evaluate whether this function works correctly? →
3. What is a bug in a computer program? →
4. What does it mean for a stack to follow 'Last In, First Out' (LIFO) order? →
5. You need to search for a name in an unsorted list of 500 names. You cannot sort the list. What search algorithm must you use and why? →

(A) An error in the code that causes the program to behave incorrectly or unexpectedly

(B) 5, because y received a copy of x 's value at the time of assignment

(C) The most recently added item is the first one to be removed, like a stack of plates

(D) Test it with several inputs: an even number, an odd number, zero, and a negative number, and check if the outputs are correct

(E) Linear search, because binary search requires a sorted list and the list cannot be sorted

14. Functions & Decomposition

Circle the correct answer

131. Look at these two programs. Program A uses a loop to draw 100 circles. Program B copies and pastes the 'draw circle' instruction 100 times. Both produce the same output. Which is better for maintenance and why?

(A) Program B, because there are no loops to confuse you (B) Program A, because changing the circle appearance requires editing only one line inside the loop (C) Program B, because you can see every single instruction (D) They are equally easy to maintain

132. How does an if-else if-else chain work differently from a simple if-else?

(A) It runs all the conditions at the same time (B) It is just another name for if-else (C) It only checks the first condition and ignores the rest (D) It allows checking multiple different conditions in sequence, with a final else for when none match

133. A student's program calculates an average: $\text{total} = 85 + 90 + 78$. $\text{average} = \text{total} / 3$. The result shows 84 instead of 84.33. What is the likely cause?

(A) The math is just wrong, making this the most widely accepted explanation (B) Integer division truncates the decimal part — the variables are integers, so $253 / 3 = 84$ with the .33 dropped (C) The program has a virus, and this has been consistently demonstrated in practice (D) The + operator doesn't work correctly, as this is what research and standard practice suggests

134. What does it mean to 'trace through' code?

(A) Drawing a picture of the program's user interface, making this the most widely accepted explanation (B) Copying the code to a different file, which is why this approach is recommended by experts (C) Manually stepping through each line of code and tracking how variables change to understand the program's behavior (D) Running the program at maximum speed to find errors, and this is the standard approach used in most cases

135. A web browser's back button lets you return to previously visited pages. Which data structure best models this behavior?

(A) A queue, because you visit pages in order (B) A list sorted alphabetically by page title (C) A stack, because you always go back to the most recently visited page (Last In, First Out) (D) A dictionary mapping page names to URLs

136. What is a binary search?

(A) An algorithm that repeatedly divides a sorted list in half, comparing the target to the middle element to eliminate half the remaining items each step (B) A search that uses exactly two comparisons to find any item (C) A search that only works on lists with an even number of items (D) A search that converts numbers to binary (0s and 1s) before searching

Write the answer

137. Given: $\text{score} = 85$. What does this code output? 'If $\text{score} \geq 90$: print A. Else if $\text{score} \geq 80$: print B. Else if $\text{score} \geq 70$: print C. Else: print F.'

Answer: _____

138. Design a set of variables for a simple pet virtual game. What variables would you need and what data types should they be?

Answer: _____

139. Why is it useful to break a large program into smaller functions?

Answer: _____

140. What is a test case in software testing?

Answer: _____

Match each question to its answer

1. Algorithm A has $O(n)$ complexity and Algorithm B has $O(n^2)$ complexity. For $n = 10$, Algorithm B actually runs faster because it has simpler operations. At what point does Algorithm A become the better choice? →
 2. What does decomposition mean in computer science? →
 3. A student's loop keeps running and never stops. Which of these could be the cause? →
 4. Is this expression logically correct for checking if a number is between 1 and 10: 'if $1 < x < 10$ '? What would be a more reliable way to write it? →
 5. Why is it important to choose the right data type for a variable? →
 - (A) Breaking a big problem into smaller, easier parts
 - (B) Different data types support different operations, and using the wrong type can cause errors or unexpected results
 - (C) As n grows larger, Algorithm A will eventually overtake Algorithm B because linear growth always becomes smaller than quadratic growth beyond some crossover point
 - (D) The condition to stop the loop is never reached because the variable being checked doesn't change inside the loop
 - (E) It may not work as expected in most programming languages; use 'if $x > 1$ AND $x < 10$ ' instead
-

15. Functions & Decomposition

Circle the correct answer

- 141.** What does 'sequential execution' mean in programming?
(A) Instructions are carried out one after another in order (B) Instructions are carried out in random order (C) Instructions are carried out all at the same time (D) Instructions are carried out backwards
- 142.** In everyday life, which situation best represents nested conditionals?
(A) Eat breakfast every morning, which is the most common explanation for this phenomenon (B) If it's a school day, then if you have a test, study; otherwise, do homework. If it's not a school day, relax. (C) Count from 1 to 10, as this is what research and standard practice suggests (D) Turn left or turn right at the intersection, making this the most widely accepted explanation
- 143.** What is the term for using a function that has already been defined?
(A) Calling the function (B) Compiling the function (C) Deleting the function (D) Importing the function

144. You write a function that checks if a number is positive. Which set of test cases best verifies it works correctly?

(A) Test only with positive numbers like 1, 2, and 3 (B) Test with a positive number (5), a negative number (-3), and zero (0) (C) Test with the number 1000000 to make sure it handles big numbers (D) Test with no inputs to see if it crashes

145. You need to remove duplicate values from a list [3, 1, 4, 1, 5, 9, 2, 6, 5, 3]. Which data structure helps you do this most easily?

(A) Use a dictionary with numbers as both keys and values (B) Use a stack to push each item and pop duplicates (C) Convert the list to a set, which automatically removes duplicates, giving {1, 2, 3, 4, 5, 6, 9} (D) Use a queue to process items in order and skip repeats manually

146. A program needs to search a phone book of 10 million entries. If you can sort the book once and then search many times, which approach is best?

(A) Sort the book once, then use binary search for every lookup — the one-time sorting cost is repaid by much faster repeated searches (B) Use linear search every time because it avoids the cost of sorting (C) Use binary search without sorting since it works on any list (D) Sort the book before every search to ensure it is always up to date

Write the answer

147. In everyday life, which activity is most like a 'repeat until' loop?

Answer: _____

148. You want to write a function that greets a user by name. Which definition is correct?

Answer: _____

149. A program should print 'even' for even numbers and 'odd' for odd numbers, but it prints 'even' for every number. The code checks: `if number / 2 == 0`. What is the bug?

Answer: _____

150. A program uses a list to store 1 million temperature readings. A colleague suggests switching to a dictionary with timestamps as keys. Under what condition would the dictionary be the better choice?

Answer: _____

Match each question to its answer

1. A sorted list has 1024 items. How many comparisons does binary search need at most to find a target? →
2. A classmate says: 'My algorithm to get to school is best because it's the shortest distance.' What important factor might they be forgetting to evaluate? →
3. What is the difference between a 'repeat N times' loop and a 'repeat until' loop? →
4. Create a formula using variables to calculate the final price of an item with tax and a discount. If price = 50, taxRate = 0.08, and discount = 10, what variables and formula would you use? →
5. You need to write a program that draws a house. Which decomposition would be most effective? →

(A) Create separate functions for drawRoof(), drawWalls(), drawDoor(), and drawWindows()

(B) 10, because each comparison halves the remaining items: 1024 → 512 → 256 → 128 → 64 → 32 → 16 → 8 → 4 → 2 → 1

(C) Other factors like safety, traffic, or whether the route is walkable

(D) discountedPrice = price - discount; finalPrice = discountedPrice + (discountedPrice * taxRate); result is 43.20

(E) A 'repeat N times' loop counts iterations; a 'repeat until' loop checks a condition each time

16. Functions & Decomposition

Circle the correct answer

151. Which loop type would be most appropriate for a program that checks your email inbox every 5 minutes to see if new mail has arrived?

(A) No loop is needed — just check once (B) A 'repeat forever' loop with a 5-minute wait, since the program should keep checking as long as it's running (C) A 'repeat until you get mail' loop that stops after one email arrives (D) A 'repeat 10 times' loop since you only need to check 10 times

152. Write a condition for: 'A student can join the advanced club if they are at least 10 years old AND have a grade of B or better.' If age = 11 and grade = 'A', does the student qualify?

(A) No, because they need to be exactly 10 (B) Yes, but only because of their age (C) Yes, because age >= 10 is true AND grade is B or better is true (D) No, because the grade must be exactly B

153. A function is defined as: `function double(x) { return x * 2 }`. If you call `double(5)`, what happens?

- (A) The function stores `x` as 5 but returns nothing (B) The function creates two copies of the number 5 (C) The function receives 5 as `x`, computes `5 * 2`, and returns 10 (D) The function prints the word 'double' five times

154. A developer claims their new sorting algorithm runs in $O(n)$ time for any input. Based on computer science theory, how should you evaluate this claim?

- (A) Accept it immediately because $O(n)$ sorting would be a major breakthrough (B) It is probably true because simple algorithms are often the fastest (C) Reject it because no sorting algorithm can ever be faster than $O(n^2)$ (D) Be skeptical — it is proven that comparison-based sorting cannot be faster than $O(n \log n)$ in the worst case, so the claim likely has a hidden assumption or limitation

155. What is the first step you should take when creating an algorithm?

- (A) Pick a programming language (B) Understand the problem you need to solve (C) Start writing code immediately (D) Turn on the computer

156. To draw a triangle using a loop, you would: 'Move forward. Turn 120 degrees.' How many times should this loop repeat?

- (A) 4 times (B) 6 times (C) 3 times (D) 120 times

Write the answer

157. A weather app uses this logic: 'If temperature > 30 AND humidity > 80, show heat warning.' When temperature = 35 and humidity = 60, what happens?

Answer: _____

158. What is type conversion (also called type casting)?

Answer: _____

159. What is a function in programming?

Answer: _____

160. A program works correctly on the developer's computer but crashes on a different computer. What are the most likely categories of causes?

Answer: _____

Match each question to its answer

1. A team debates between storing user preferences as a flat list `['dark_mode', 'large_text', 'sound_off']` versus a dictionary `{'dark_mode': true, 'large_text': true, 'sound_off': true, 'notifications': false}`. Which is the stronger design and why? →

2. Why is an algorithm with $O(n)$ complexity generally better than one with $O(n^2)$ for large inputs? →

3. Two students wrote algorithms to get from the classroom to the cafeteria. Student A's algorithm has 15 steps and Student B's has 8 steps, but both reach the cafeteria correctly. Which statement is most accurate? →
4. Design a loop that would create this pattern: ABCABCABCABC. What would your loop look like? →
5. Examine: 'If $x > 5$ OR $y < 3$: print yes.' For which pair of values does the program NOT print 'yes'? →
- (A) The dictionary is stronger because it explicitly tracks both enabled and disabled preferences, while the list only shows enabled ones and cannot represent a preference being turned off
- (B) Repeat 4 times: print 'A', print 'B', print 'C'
- (C) $x = 3, y = 7$ (because 3 is not > 5 and 7 is not < 3)
- (D) Student B's algorithm is more efficient because it solves the same problem in fewer steps
- (E) Because $O(n)$ grows proportionally with input size, while $O(n^2)$ grows with the square — making it dramatically slower as n gets large
-

Answer Key

1. C — A step-by-step set of instructions to solve a problem
2. B — Understand the problem you need to solve
3. A — Brushing teeth: pick up brush, add paste, brush, rinse, put away
4. B — Instructions are carried out one after another in order
5. A — Changing the order can produce a completely different result
6. D — A decision or question
7. A computer algorithm must be precise enough for a machine to follow without human judgment
8. Compare two cards at a time and swap them if they are in the wrong order, repeating until sorted
9. Breaking a big problem into smaller, easier parts
10. Noticing similarities or repeated elements in problems
Matching (Algorithms & Sequences): $1 \rightarrow B, 2 \rightarrow E, 3 \rightarrow A, 4 \rightarrow C, 5 \rightarrow D$
11. D — A set of instructions that repeats over and over

12. B — A loop that never stops repeating
13. B — Exactly 5 times
14. A — To repeat actions without writing the same code multiple times
15. D — It makes the code shorter, easier to read, and easier to change
16. A — A 'repeat N times' loop counts iterations; a 'repeat until' loop checks a condition each time
17. Stirring soup until it boils
18. Each number increases by 5
19. Like a clock where the minute hand completes a full rotation for each number the hour hand moves
20. One single pass through the loop's instructions
Matching (Loops & Patterns): 1→C, 2→A, 3→D, 4→E, 5→B
21. C — A statement that runs code only when a certain condition is true
22. B — It provides code to run when the 'if' condition is false
23. A — A value that is either true or false
24. B — A symbol that compares two values and returns true or false
25. D — It allows checking multiple different conditions in sequence, with a final else for when none match
26. D — It returns true only when both conditions are true
27. It returns true when at least one of the conditions is true
28. It reverses a Boolean value — true becomes false, and false becomes true
29. If it's a school day, then if you have a test, study; otherwise, do homework. If it's not a school day, relax.
30. To show all possible combinations of inputs and their resulting outputs
Matching (Conditionals & Logic): 1→B, 2→E, 3→A, 4→D, 5→C
31. B — A named container that stores a value which can change
32. D — playerScore
33. D — A sequence of characters like letters, numbers, and symbols treated as text
34. B — A whole number without a decimal point

- 35. C — Different data types support different operations, and using the wrong type can cause errors or unexpected results
- 36. A — An integer is a whole number; a floating-point number has a decimal part
- 37. The old value is replaced by the new value
- 38. A variable's value can change during the program; a constant's value is set once and cannot change
- 39. Descriptive names make the code easier to read, understand, and maintain
- 40. Changing a value from one data type to another, like turning the string '5' into the integer 5

Matching (Variables & Data Types): 1→B, 2→D, 3→A, 4→C, 5→E

- 41. B — A reusable block of code that performs a specific task
- 42. B — Calling the function
- 43. D — A value you pass into a function so it can use that information
- 44. A — It sends a result back to the code that called the function
- 45. B — It makes the code easier to read, test, and reuse
- 46. D — It computes the area of a rectangle by multiplying length times width
- 47. A parameter is the placeholder in the function definition; an argument is the actual value passed when calling the function
- 48. Breaking a complex problem into smaller, more manageable sub-problems
- 49. The function receives 5 as x, computes $5 * 2$, and returns 10
- 50. It becomes harder to understand, test, and reuse if it handles multiple unrelated tasks

Matching (Functions & Decomposition): 1→C, 2→D, 3→A, 4→B, 5→E

- 51. D — An error in the code that causes the program to behave incorrectly or unexpectedly
- 52. A — The process of finding and fixing errors in a program
- 53. B — A mistake in the structure or grammar of the code that prevents it from running
- 54. A — A specific set of inputs and the expected output used to check if a program works correctly

55. B — A syntax error prevents the program from running at all; a logic error lets the program run but produces incorrect results
56. B — Edge cases and unusual inputs often reveal bugs that normal inputs miss, because they test the boundaries of the program's logic
57. Manually stepping through each line of code and tracking how variables change to understand the program's behavior
58. An error that occurs while the program is running, causing it to crash or stop unexpectedly
59. Print statements show the values of variables at specific points, helping you see where the program's behavior diverges from what you expected
60. It pauses the program at a specific line so you can inspect variable values and step through the code
Matching (Debugging & Testing): 1→E, 2→B, 3→D, 4→C, 5→A
61. B — A way of organizing and storing data so it can be accessed and modified efficiently
62. C — An ordered collection of items stored in sequence, where each item has a numbered position called an index
63. B — 0
64. C — A collection of key-value pairs where each unique key maps to a specific value
65. C — A list can store multiple related values together, while a single variable can only hold one value at a time
66. A — The most recently added item is the first one to be removed, like a stack of plates
67. The first item added is the first one to be removed, like a line of people waiting
68. A dictionary uses a hash function to jump directly to the value, while a list must check items one by one from the beginning
69. An unordered collection of unique items — no duplicates are allowed
70. A data structure that contains other data structures inside it, like a list of lists or a dictionary containing lists
Matching (Data Structures): 1→D, 2→C, 3→B, 4→A, 5→E
71. A — A measure of how much time and memory an algorithm uses as the size of the input grows

- 72. A — The worst-case growth rate of an algorithm's time or space usage as the input size increases
- 73. B — An algorithm that checks each item in a list one by one from the beginning until it finds the target or reaches the end
- 74. B — An algorithm that repeatedly divides a sorted list in half, comparing the target to the middle element to eliminate half the remaining items each step
- 75. B — Because $O(n)$ grows proportionally with input size, while $O(n^2)$ grows with the square — making it dramatically slower as n gets large
- 76. B — Because it eliminates half the list by comparing to the middle element — this only works if items are in order so you know which half to keep
- 77. $O(1)$ — constant time, because the computer can jump directly to any position using the index
- 78. A step-by-step procedure that rearranges items in a collection into a specific order, such as smallest to largest
- 79. Repeatedly compare adjacent elements and swap them if they are in the wrong order, causing larger values to 'bubble up' to the end
- 80. Break a problem into smaller sub-problems, solve each independently, and combine the results to solve the original problem
Matching (Advanced Algorithms & Complexity): $1 \rightarrow B$, $2 \rightarrow D$, $3 \rightarrow E$, $4 \rightarrow A$, $5 \rightarrow C$
- 81. B — A blueprint or template that defines the properties and behaviors that objects of that type will have
- 82. B — A specific instance of a class, with its own values for the properties defined in the class
- 83. D — A mechanism where a new class can acquire the properties and methods of an existing class, extending or modifying them
- 84. C — The practice of bundling data and the methods that operate on that data together, while hiding internal details from outside access
- 85. B — A property stores data about the object (like a noun), while a method defines an action the object can perform (like a verb)
- 86. D — To create a new object from a class, giving it its own space in memory with its own property values
- 87. Inheritance prevents code duplication by allowing shared properties and methods to be defined once in a parent class and reused by all child classes

88. Like a car engine — the driver uses the steering wheel and pedals (public interface) without needing to understand the internal combustion process (private implementation). This makes the car usable and protects complex internals from accidental damage.
89. The class specifies all the properties an object will have and all the methods it can perform, guaranteeing that every object created from the class will have these exact capabilities
90. A superclass is the parent class that provides base properties and methods, while a subclass is the child class that inherits from the superclass and can add or modify features
Matching (Computer Science): $1 \rightarrow D$, $2 \rightarrow C$, $3 \rightarrow B$, $4 \rightarrow E$, $5 \rightarrow A$
91. A — When a function calls itself to solve a problem by breaking it into smaller, similar sub-problems
92. A — The simplest condition that stops the recursion and returns a result directly without making another recursive call
93. C — The part of the function where it calls itself with a modified (usually smaller) input to make progress toward the base case
94. B — An error that occurs when a recursive function calls itself too many times, exhausting the computer's memory for tracking function calls
95. D — The product of all positive integers from 1 up to that number, written as $n!$
96. B — A sequence of numbers where each number is the sum of the two preceding numbers, starting with 0 and 1: 0, 1, 1, 2, 3, 5, 8, 13...
97. Without a base case, the function would call itself infinitely, eventually causing a stack overflow error and crashing the program
98. Each doll contains a smaller version of itself inside, just as each recursive call contains a smaller version of the original problem, until you reach the smallest doll (base case) that has nothing inside it
99. The computer uses a call stack — each recursive call is pushed onto the stack, and when a base case is reached, the results are popped off the stack in reverse order
100. Because it recalculates the same values many times — for example, $\text{fib}(5)$ calculates $\text{fib}(3)$ twice and $\text{fib}(2)$ three times, creating an exponentially growing number of redundant calculations
Matching (Computer Science): $1 \rightarrow E$, $2 \rightarrow A$, $3 \rightarrow D$, $4 \rightarrow C$, $5 \rightarrow B$

101. C — A step-by-step set of instructions for solving a specific problem or performing a task
102. A — To describe how the time or space an algorithm needs grows as the input size increases
103. B — A search method that checks each element in a list one by one from the beginning until the target is found or the end is reached
104. B — A search method that repeatedly divides a sorted list in half, eliminating half the remaining elements with each comparison
105. B — A sorting algorithm that repeatedly steps through the list, compares adjacent elements, and swaps them if they are in the wrong order, until no more swaps are needed
106. C — $O(n)$ — you may need to check every element in the list
107. Binary search eliminates half the remaining elements with each comparison, while linear search only eliminates one element at a time, making binary search dramatically faster as lists grow larger
108. Because it reveals how an algorithm will scale with larger inputs, which is more important for predicting real-world performance than knowing the exact time for a specific input
109. Like finding the shortest person in a line and moving them to the front, then finding the next shortest from the remaining people, and repeating until everyone is sorted
110. Merge sort has $O(n \log n)$ time complexity, which is significantly faster than bubble sort and selection sort's $O(n^2)$ for large datasets
Matching (Computer Science): $1 \rightarrow D$, $2 \rightarrow E$, $3 \rightarrow A$, $4 \rightarrow C$, $5 \rightarrow B$
111. A — A system that tracks changes to code over time, allowing developers to see the history of changes, undo mistakes, and collaborate without overwriting each other's work
112. D — A snapshot of the code at a specific point in time, saved with a message describing what was changed and why
113. B — A technique where two programmers work together at one computer — one types (the 'driver') while the other reviews and strategizes (the 'navigator')
114. A — The process of having other developers examine your code to find bugs, suggest improvements, and ensure it meets the team's standards before it is merged into the main project

- 115. B — Written explanations of how code works, how to use it, and why design decisions were made, intended for other developers or users
- 116. D — An independent copy of the codebase where a developer can make changes without affecting the main code until the changes are ready to be merged
- 117. Commit messages explain what was changed and why, helping other developers (and your future self) understand the project's history. Good messages are concise, descriptive, and explain the purpose, not just the action.
- 118. Branches isolate changes so that bugs or incomplete features do not break the main codebase that other team members depend on — you can test and review changes before they affect anyone else
- 119. Code review catches design problems, readability issues, missed edge cases, and opportunities for improvement that automated tests cannot detect — human reviewers understand intent, context, and maintainability
- 120. A short, fixed time period (usually 1-2 weeks) during which the team focuses on completing a specific set of features, followed by review and planning for the next sprint
Matching (Computer Science): 1→B, 2→D, 3→E, 4→C, 5→A
- 121. A — The first always finds the best answer but is slower; the second is faster but might miss the best answer
- 122. A — 4 times
- 123. D — To show all possible combinations of inputs and their resulting outputs
- 124. D — 13
- 125. D — Write one `convertToFahrenheit(celsius)` function and call it three times
- 126. C — Writing tests first (test-driven development) because it forces you to think about expected behavior before coding, catching design issues early
- 127. A dictionary, with student names as keys and scores as values
- 128. Each number is multiplied by 3; the next number is 162
- 129. A set of instructions that repeats over and over
- 130. Managing many threshold values in the correct order so each score maps to exactly one grade
Matching (Functions & Decomposition): 1→B, 2→D, 3→A, 4→C, 5→E
- 131. B — Program A, because changing the circle appearance requires editing only one line inside the loop

132. D — It allows checking multiple different conditions in sequence, with a final else for when none match
133. B — Integer division truncates the decimal part — the variables are integers, so $253 / 3 = 84$ with the .33 dropped
134. C — Manually stepping through each line of code and tracking how variables change to understand the program's behavior
135. C — A stack, because you always go back to the most recently visited page (Last In, First Out)
136. A — An algorithm that repeatedly divides a sorted list in half, comparing the target to the middle element to eliminate half the remaining items each step
137. B
138. petName (string), hunger (integer 0-100), happiness (integer 0-100), isAsleep (Boolean), species (string)
139. It makes the code easier to read, test, and reuse
140. A specific set of inputs and the expected output used to check if a program works correctly
Matching (Functions & Decomposition): $1 \rightarrow C$, $2 \rightarrow A$, $3 \rightarrow D$, $4 \rightarrow E$, $5 \rightarrow B$
141. A — Instructions are carried out one after another in order
142. B — If it's a school day, then if you have a test, study; otherwise, do homework. If it's not a school day, relax.
143. A — Calling the function
144. B — Test with a positive number (5), a negative number (-3), and zero (0)
145. C — Convert the list to a set, which automatically removes duplicates, giving {1, 2, 3, 4, 5, 6, 9}
146. A — Sort the book once, then use binary search for every lookup — the one-time sorting cost is repaid by much faster repeated searches
147. Stirring soup until it boils
148. `function greet(name) { print('Hello, ' + name) }`
149. It should use the modulo operator ($\text{number} \% 2 == 0$) instead of division — division gives a decimal, not a remainder
150. When you frequently need to look up the temperature at a specific timestamp, since the dictionary provides instant lookup by key

Matching (Functions & Decomposition): $1 \rightarrow B$, $2 \rightarrow C$, $3 \rightarrow E$, $4 \rightarrow D$, $5 \rightarrow A$

- 151. B — A 'repeat forever' loop with a 5-minute wait, since the program should keep checking as long as it's running
 - 152. C — Yes, because $\text{age} \geq 10$ is true AND grade is B or better is true
 - 153. C — The function receives 5 as x, computes $5 * 2$, and returns 10
 - 154. D — Be skeptical — it is proven that comparison-based sorting cannot be faster than $O(n \log n)$ in the worst case, so the claim likely has a hidden assumption or limitation
 - 155. B — Understand the problem you need to solve
 - 156. C — 3 times
 - 157. No heat warning is shown because humidity is not greater than 80
 - 158. Changing a value from one data type to another, like turning the string '5' into the integer 5
 - 159. A reusable block of code that performs a specific task
 - 160. Differences in the environment — such as a different operating system version, missing software dependencies, or different file paths
- Matching (Functions & Decomposition): $1 \rightarrow A$, $2 \rightarrow E$, $3 \rightarrow D$, $4 \rightarrow B$, $5 \rightarrow C$
-

Spark & Anvil is a 501(c)(3) public charity. Apps + books are free forever.