## JavaScript Logical Solving Questions

This document contains 15 intermediate-level questions designed to help you practice and understand fundamental JavaScript control statements and logical problem-solving. All solutions should be implemented for the console environment only.

---

## Section 1: Conditional Statements (if, else if, else, switch)

1. **Grade Calculator** Write a JavaScript program that prompts a user for a numerical grade (0-100). Use `if-else if-else` to log the corresponding letter grade to the console:
    - 90-100: "A"
    - 80-89: "B"
    - 70-79: "C"
    - 60-69: "D"
    - Below 60: "F"
2. **Simple ATM Simulation**: Simulate a simple ATM. Start with a variable `balance` set to 500. Prompt the user for a transaction type: 'deposit', 'withdraw', or 'check balance'. Use a `switch` statement to handle the following:
    - If 'deposit', prompt for an amount and add it to the `balance`.
    - If 'withdraw', prompt for an amount. Check if the `balance` is sufficient; if so, subtract the amount. Otherwise, log "Insufficient funds.".
    - If 'check balance', log the current `balance`.
    - For any other input, log "Invalid transaction type.".
    - After each transaction, log the new `balance`.
3. **Leap Year Checker** Write a program that prompts the user for a year. Use `if-else if-else` statements to determine if it's a leap year. A leap year is divisible by 4, but not by 100 unless it is also divisible by 400. Log "Leap Year" or "Not a Leap Year".

---

## Section 2: Loops (for, while, do-while)

4. **FizzBuzz Challenge**: Write a `for` loop that iterates from 1 to 100.
    - For multiples of 3, log "Fizz".
    - For multiples of 5, log "Buzz".
    - For numbers that are multiples of both 3 and 5, log "FizzBuzz".
    - For all other numbers, log the number itself.

5. **Vowel Counter**: Create a function that takes a string as input. Use a `for` loop to iterate through the string and count the number of vowels (a, e, i, o, u, case-insensitive). Log the final count to the console.
6. **Factorial Calculator** Write a JavaScript program that calculates the factorial of a given number (e.g., `5!` is `5 * 4 * 3 * 2 * 1`). Prompt the user for a number and use a `for` loop to compute and log the result.
7. **Countdown Timer**: Use a `while` loop to create a countdown from a user-specified number to 0. Log each number to the console. After the loop, log "Blastoff!".
8. **Guess the Number Game:** Create a simple "Guess the Number" game. Generate a random number between 1 and 100. Use a `do-while` loop to repeatedly prompt the user for a guess.
    - If the guess is too high, log "Too high! Try again.".
    - If the guess is too low, log "Too low! Try again.".
    - The loop should continue until the user guesses the correct number, at which point you should log "Congratulations! You guessed the number.".
9. **Sum of Digits**: Create a program that takes a number as input and calculates the sum of its digits. Use a `while` loop to extract and sum each digit. For example, the sum of digits for 123 is 1 + 2 + 3 = 6.
10. **Multiplication Table Generator** Prompt the user for a number. Use a `for` loop to generate and log the multiplication table for that number from 1 to 10. The output should be in the format: "5 x 1 = 5", "5 x 2 = 10", etc.
11. **Fibonacci Sequence** Write a program that generates and logs the first 10 numbers of the Fibonacci sequence. The sequence starts with 0 and 1, and each subsequent number is the sum of the two preceding ones (e.g., 0, 1, 1, 2, 3, 5...). Use a `for` loop.

---

## Section 3: Combined Logic and Loop Control

12. **Prime Number Checker** Write a program that prompts the user for a number. Use a `for` loop and the `break` statement to determine if the number is prime. A prime number is a natural number greater than 1 that is not a product of two smaller natural numbers. Log "The number is prime." or "The number is not prime."
13. **Palindrome Checker** Write a function that checks if a given string is a palindrome (reads the same forwards and backward, e.g., "racecar"). Use a `for` loop to iterate and compare characters from both ends of the string. The function should log `true` or `false`.
14. **Find the Largest Number in an Array** Given an array of numbers, write a program to find the largest number. Use a `for` loop to iterate through the array and keep track of the maximum value found so far. Log the final largest number.

15. **Odd or Even Counter with `continue`** Write a `for` loop that iterates from 1 to 20. Use the `continue` statement to skip all odd numbers. The loop should only log the even numbers to the console.