**1. Prime Numbers**

* **Count Primes** (LeetCode #204): Count the number of prime numbers less than a non-negative number n.
* **Ugly Number** (LeetCode #263): Determine if a number is an "ugly number" (divisible only by 2, 3, or 5).

**2. Number Theory**

* **Greatest Common Divisor of Strings** (LeetCode #1071): Find the largest string X such that X concatenated some number of times equals both input strings.
* **Integer to Roman** (LeetCode #12): Convert an integer to its Roman numeral representation.
* **Roman to Integer** (LeetCode #13): Convert a Roman numeral string to an integer.

**3. Modular Arithmetic**

* **Power of Three** (LeetCode #326): Check if a number is a power of three.
* **Pow(x, n)** (LeetCode #50): Implement a function to calculate x raised to the power n (fast exponentiation).
* **Add Binary** (LeetCode #67): Add two binary strings and return their sum as a binary string.

**4. Bit Manipulation**

* **Single Number** (LeetCode #136): Find the single number that does not appear twice in an array.
* **Hamming Distance** (LeetCode #461): Find the number of differing bits between two integers.
* **Number of 1 Bits** (LeetCode #191): Count the number of 1 bits in the binary representation of a number.

**5. Mathematical Properties**

* **Happy Number** (LeetCode #202): Determine if a number eventually reaches 1 after repeatedly replacing it with the sum of the squares of its digits.
* **Valid Perfect Square** (LeetCode #367): Check if a number is a perfect square without using the square root function.
* **Fizz Buzz** (LeetCode #412): Simple problem of printing "Fizz," "Buzz," or "FizzBuzz" for multiples of 3, 5, or both.

**6. Arrays and Mathematics**

* **Maximum Product Subarray** (LeetCode #152): Find the contiguous subarray within an array that has the largest product.
* **Majority Element** (LeetCode #169): Find the majority element that appears more than n/2 times in an array.
* **Pascal’s Triangle** (LeetCode #118): Generate the first numRows of Pascal's Triangle.

**7. Combinatorics**

* **Climbing Stairs** (LeetCode #70): Count the number of distinct ways to climb a staircase of n steps (can take 1 or 2 steps at a time).
* **Combination Sum** (LeetCode #39): Find all unique combinations of numbers that sum up to a target.

**8. Geometry**

* **Valid Boomerang** (LeetCode #1037): Determine if three points form a boomerang (a non-collinear triangle).
* **Rectangle Overlap** (LeetCode #836): Check if two rectangles overlap.

**9. Miscellaneous**

* **Sqrt(x)** (LeetCode #69): Implement the square root function.
* **Factorial Trailing Zeroes** (LeetCode #172): Find the number of trailing zeroes in the factorial of a number.
* **Multiply Strings** (LeetCode #43): Multiply two large numbers represented as strings.

**10. Simulation Problems**

* **Spiral Matrix** (LeetCode #54): Traverse a 2D matrix in a spiral order.
* **Rotate Image** (LeetCode #48): Rotate a 2D matrix by 90 degrees (clockwise).

**Tips for Preparation:**

1. **Understand Basic Concepts**: Know prime factorization, modular arithmetic, and Euclidean algorithms.
2. **Practice Edge Cases**: Think about large inputs, negative numbers, and zeros.
3. **Master Fast Algorithms**: Learn optimized approaches like fast exponentiation and bit manipulation.
4. **Analyze Complexity**: Always aim for the most efficient solution.