Bit Manipulation Programming Questions in C

Basic Questions

1. Set a bit at position n in an integer.
2. Clear a bit at position n in an integer.
3. Toggle a bit at position n in an integer.
4. Check if the nth bit is set in a number.
5. Count the number of 1s in a binary representation of a number.
6. Check if a number is even or odd using bitwise operators.
7. Multiply a number by 2 using bitwise shift.
8. Divide a number by 2 using bitwise shift.
9. Check if a number is a power of 2.
10. Swap two numbers without using a temporary variable.

Intermediate Questions

1. Find the only non-repeating element in an array where every other element appears twice.
2. Find the two non-repeating elements in an array where every other element appears twice.
3. Turn off the rightmost set bit.
4. Turn on the rightmost 0 bit.
5. Get the position of the rightmost set bit.
6. Check if two integers have opposite signs.
7. Add 1 to an integer using bitwise operators (no + allowed).
8. Subtract 1 from an integer using bitwise operators (no - allowed).
9. Find the binary representation of a number using bitwise.
10. Determine if the ith bit in a number is set (0 or 1).

Advanced Questions

1. Reverse the bits of a 32-bit unsigned integer.
2. Find the number that appears once in an array where every element appears thrice.

3. Implement addition, subtraction, multiplication, and division using only bitwise operators.

4. Calculate XOR of all numbers from 1 to n.

5. Find XOR of two numbers without using ^ operator.

6. Gray to Binary and Binary to Gray code conversion.

7. Find the next higher and lower number with the same number of set bits.

8. Count total bits needed to convert a number A to B.

9. Copy bits from one number to another in a specific position.

10. Insert bits of M into N from position i to j.