Power of Two

- Given an integer
- Check if it is a power of two
  - No loops, no recursion, no floating-point functions

A Single

- Given a long sequence of N integers
- Every value occurs twice, except for one, which occurs once
- Find the single value
  - O(1) memory, O(N) time

Two Singles

- Given a long sequence of N integers
- Every value occurs twice, except for two, which occur once each
- Find the two single values
  - O(1) memory, O(N) time

Maximal XOR

- Given integers $L \leq R$
- Find maximal possible A xor B where $L \leq A \leq B \leq R$

Sum or XOR

- Given an integer N
- Find the number of $0 < X < N$ such that $N \text{ xor } X = N + X$

Greater XOR

- Given an integer N
- Find the number of $0 < X < N$ such that $N \text{ xor } X > N$

XOR Sequence

- Define sequence A
  - A[0] = 0
  - A[k] = A[k-1] xor k
- Given integers $L \leq R$
- Find A[L] xor A[L+1] xor … xor A[R]

Testing

- https://www.hackerrank.com/challenges/lonely-integer
- https://www.hackerrank.com/challenges/maximizing-xor
- https://www.hackerrank.com/challenges/sum-vs-xor
- https://www.hackerrank.com/challenges/the-great-xor
- https://www.hackerrank.com/challenges/xor-se