Bitwise XOR (eXclusive OR)

The **XOR** bitwise operation (or **exclusive or**) takes two sets of bits, and for each pair (the two bits at the same index in each bit set) returns 1 only if **one but not both** of the bits is 1. Otherwise, it returns 0.

```java
1 ^ 1 // 0
1 ^ 0 // 1
0 ^ 1 // 1
0 ^ 0 // 0
```

Think of it like a bag of chips where only one hand can fit in at a time. If no one reaches for chips, no one gets chips, and if both people reach for chips, they can't fit and no one gets chips either!

When performing XOR on two integers, only digit columns used by **one but not both** integers remain:

```java
5 ^ 6 // Gives 3


// At the bit level:
//     0101  (5)
//   ^ 0110  (6)
//   = 0011  (3)
```

See also:

- Binary Numbers (/concept/binary-numbers)
- Bitwise AND (/concept/and)
- Bitwise OR (/concept/or)
- Bitwise NOT (/concept/not)
- Bit Shifting (/concept/bit-shift)

Bitwise XOR (eXclusive OR) Coding Interview Questions

21✔ **The Stolen Breakfast Drone »**

In a beautiful Amazon utopia where breakfast is delivered by drones, one drone has gone missing. Write a function to figure out which one is missing. keep reading »

(/question/find-unique-int-among-duplicates)

All Questions ➜ (/all-questions)
