

# Goal

- Today's goal is to take a look at bit manipulation.
- You already had a taste of it while generating subsets.
- Now we will look at a few helpful builtin functions:
  - `__builtin_clz(x)`
  - `__builtin_ctz(x)`
  - `__builtin_popcount(x)`
  - `__builtin_parity(x)`
- We will also be taking a look at the bitset data structure.

# Resources

- <https://youtu.be/xXKL9YBWgCY?si=U8zcmJtnDUSa17Sg>
  - This is a great video by Errichto giving an introduction bit manipulation.
- <https://youtu.be/jqJ5s0770Ko?si=EG1LZAjN3xmtmn4c>
  - This is another video by Errichto explaining about bitsets.
- <https://usaco.guide/CPH.pdf#page=108>
  - Tell us about the builtin bitwise functions.
- Important Note
  - When you need to allocate very large chunks of contiguous memory like 128 MB it can cause a stack overflow.
  - In such cases declaring it globally will change the allocation from the stack to the process' data segment which can handle such large contiguous blocks of memory.

# Problems

1. <https://cses.fi/problemset/task/1621/>
  - Solve this using bitsets.
2. <https://cses.fi/problemset/task/2185/>
  - Hint 1: Iterate through all subsets of the primes.
  - Hint 2: For each subset of primes count the number of number divisible by all in the subset.
3. [https://atcoder.jp/contests/abc295/tasks/abc295\\_d](https://atcoder.jp/contests/abc295/tasks/abc295_d)
  - Hint 1: You need to find subarrays where each digit occurs an even number of times.
  - Hint 2: You can create a bitmask for the digits.
  - Hint 3: Each time a digit occurs flip is corresponding bit.