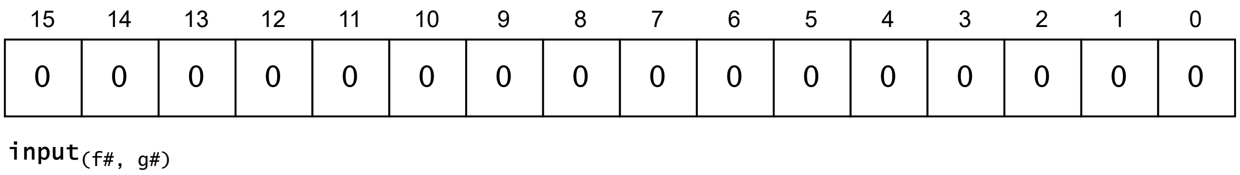
Bloom Filter through an example

Problem: we want to check if an item has already been added to a list. (Items are stored in another data structure.) We want to avoid searching through the items since it may take a long time, and also we would like to use minimal resources, like memory or storage. The solution is bloom filter, which will let us know if an item has already been added or not. This is a probabilistic data structure, so it may generate false positives but will never generate false negatives.

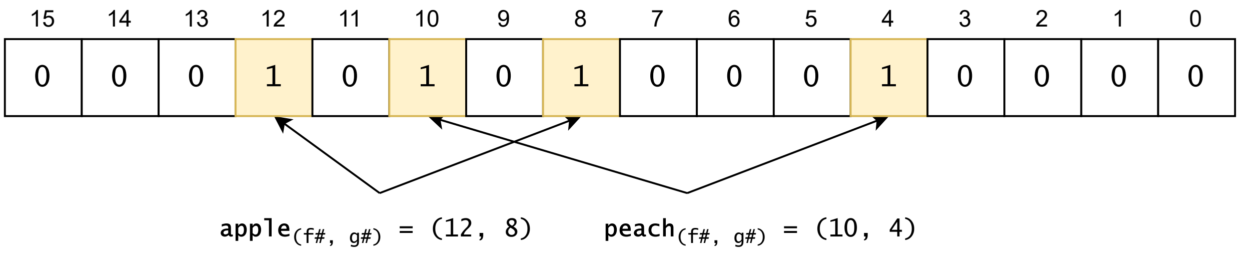
Step 0. Initialization

→ **Starting with an empty bitmap, and two hash functions (f, g).**



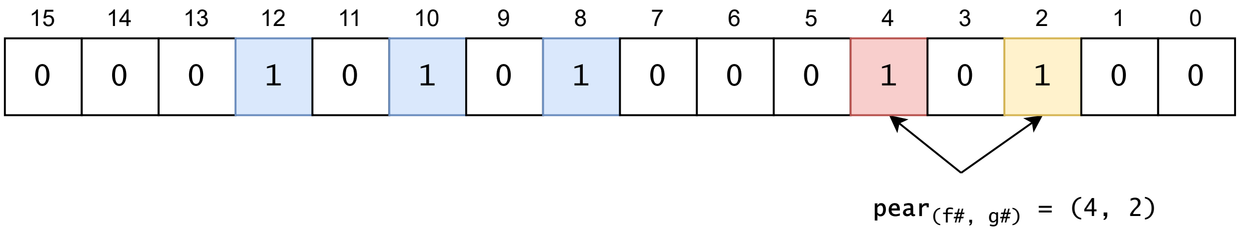
Step 1. Adding few elements by turning their bits to 1.

→ **Vanilla case, no collisions, items can be added.**



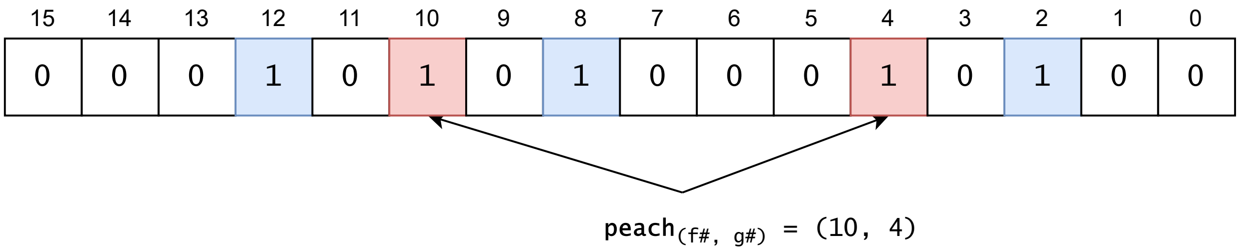
Step 2. Adding a partially colliding element

→ **Can be added, since collision is only partial.**



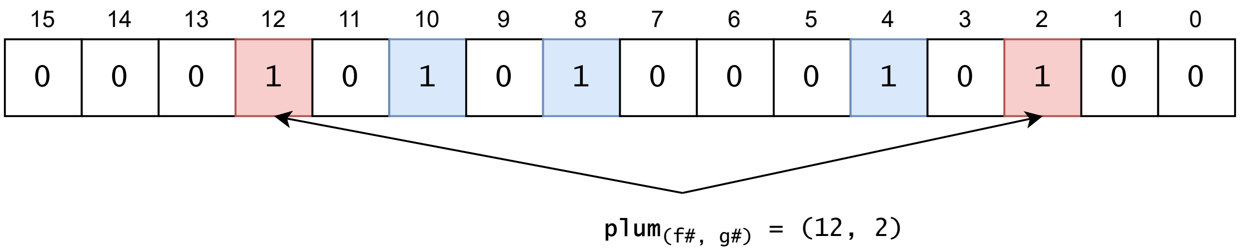
Step 3. Trying to add an already added item

→ **Collision, as expected.**



Step 4. Trying to add a new item with false positive result

→ **Collision, although item wasn't added yet.**



In the end the bloom filter is actually a (usually huge) number, like in this example it is **5396**.