

1. Suppose that you are given the bitmap representation of a set  $S$  of non-negative integers. Describe how to find the smallest non-negative integer  $x$  that is *not* a member of  $S$ , in a constant number of integer operations.

(This is an important operation in combinatorial game theory, where it is called the “minimum excluded value” or “mex”. You may assume that the dictionary `set2element` from the lecture notes has already been computed and that looking up a key-value pair in that dictionary is allowed among the constant number of operations that you perform. You may also assume that the resulting value  $x$  is within the range of values allowed by your bitmap set representation.)

■

2. The lecture notes provide formulas that, for bitmap-based sets, perform the operations of set union, set intersection, and testing whether one set is a subset of another. Which of these formulas work correctly without modification on Bloom filters? (You can assume that the Bloom filter is small enough to fit into a single word of memory, so that the same formulas can be used; the question is about the correctness of these formulas, not about their implementation.) Explain your answers.

■

3. We have seen that, to achieve a false positive rate of  $\varepsilon$  for a set of  $n$  items, Bloom filters need  $(n \log_2 1/\varepsilon)/\log 2$  bits of storage. We didn’t analyze cuckoo filters as precisely, but many sources give a bound of  $n(2 + \log_2 1/\varepsilon)$  on the number of bits of storage needed for them. Assuming that this bound is accurate, for which values of  $\varepsilon$  do cuckoo filters use fewer bits than Bloom filters, and for which values of  $\varepsilon$  do cuckoo filters use more bits? (It may be easiest to solve this numerically by writing a short computer program.)

■

4. The lectures for this week include a proof that, if the strong exponential time hypothesis (SETH) is true, then for every constant  $\varepsilon > 0$  the disjointness data structure problem for  $N$  sets of size  $k$  cannot be solved with preprocessing time  $N^{2-\varepsilon}k^{O(1)}$  and query time  $N^{1-\varepsilon}k^{O(1)}$ . Prove a stronger bound on the preprocessing, with the same bound on the query time: if SETH is true, then for every two constants  $a > 0$  and  $\varepsilon > 0$ , the disjoint set data structure problem cannot be solved with preprocessing time  $N^a k^{O(1)}$  and query time  $N^{1-\varepsilon}k^{O(1)}$ .

Hint: Follow the same proof strategy as in the lecture notes, but partition the variables of the satisfiability instance into two subsets of unbalanced sizes, depending on  $a$ . Which collection of partial truth assignments should be used as the data for the disjointness structure, and which collection of partial truth assignments should be used as the queries?

■