

## Exercises, chapter 13, solutions

---

### Answer 1

By assumption (i), each of the  $n$  keys is equally likely to hash into any of the  $m$  slots, so the expected length of every list in  $T$  is  $n/m$  (see below for a proof). To search unsuccessfully for a key  $k$  means that we search to the end of the list  $T[h]$ , where  $h$  is the hash value of  $k$ . By assumption (ii), computing  $h$  takes  $O(1)$  time. Thus, in total, the expected time for an unsuccessful search is  $\Theta(1 + n/m)$ .

#### Remark for students who are interested in probability theory:

For each  $h \in \{1, 2, \dots, m\}$ , the length of the list  $T[h]$  equals the number of keys that are hashed to slot  $h$  by **Hash**. Therefore, to obtain the expected length of  $T[h]$  for any fixed  $h$ , we let  $X$  be a random variable that represents the number of keys that are hashed to slot  $h$  by **Hash** and compute  $E[X]$  as follows.

For each  $i \in \{1, 2, \dots, n\}$ , define an indicator random variable  $X_i$  such that  $X_i$  is 1 if the  $i$ th key is hashed to slot  $h$  by **Hash** and 0 otherwise. According to the definition of “expected value”,

$$\begin{aligned} E[X_i] &= \sum_j j \cdot \Pr\{X_i = j\} = 0 \cdot \Pr\{X_i = 0\} + 1 \cdot \Pr\{X_i = 1\} + 0 + 0 + \dots = \\ &= 0 \cdot \Pr\{\text{Hash}(k_i) \neq h\} + 1 \cdot \Pr\{\text{Hash}(k_i) = h\} = \Pr\{\text{Hash}(k_i) = h\} \end{aligned}$$

where  $\{k_1, k_2, \dots, k_n\}$  are the  $n$  keys stored in  $T$ . Assumption (i) says that  $\Pr\{\text{Hash}(k_i) = h\} = \frac{1}{m}$ , which means  $E[X_i] = \frac{1}{m}$ .

Since  $X = \sum_{i=1}^n X_i$ , linearity of expectation gives  $E[X] = E[\sum_{i=1}^n X_i] = \sum_{i=1}^n E[X_i] = \sum_{i=1}^n \frac{1}{m} = \frac{n}{m}$ .

### Answer 2

See Chapter 13.6 in the textbook or the slides from Lecture 11.

### Answer 3(a)

The answer is no. For an example where  $T_3 \neq T_4$ , define the Bloom filters to have  $m = 5$  and to use the two hash functions  $h_0(x) = x \bmod 5$  and  $h_1(x) = 2x \bmod 5$ , and take  $S_1 = \{1\}$  and  $S_2 = \{2\}$ . Consider position 2 in the bit vectors. We have  $T_1[2] = 1$  due to  $h_1(1) = 2$  and  $T_2[2] = 1$  due to  $h_0(2) = 2$ . Since  $T_4$  is the bitwise AND between  $T_1$  and  $T_2$ ,  $T_4[2] = 1$ . However,  $S_1 \cap S_2 = \emptyset$ , so  $T_3[2] = 0$ . Thus,  $T_3 \neq T_4$ .

	0	1	2	3	4
$T_1$	.	.	1	.	.
$T_2$	.	.	1	.	.
$T_3$	0	0	0	0	0
$T_4$	.	.	1	.	.

### Answer 3(b)

As  $k$  increases, the number of bits equal to 1 in the filter increases, which contributes adversely to the false positive rate. At some point, the benefit of having additional hash functions is lost. As an extreme example, consider the case where  $k$  is so large that inserting a single element into the filter might set all bits to 1.

More formally, it can be shown that for a fixed filter size  $m$  and a fixed number of elements  $n$ , the false positive rate is approximately  $(1 - e^{-kn/m})^k$ . Since this function has a global minimum at  $k = (m/n) \ln 2$ , it is not strictly decreasing.

### Answer 3(c)

See Chapter 13.7 in the textbook or the slides from Lecture 11.