Hashing

Reset Progress Reveal Solutions

1 Hash tables

Hash tables with universal hash families guarantee an expected runtime of $O(1)$ for the INSERT, SEARCH, and DELETE operations. What is the meaning of "expected"?

○ It is an average over the choices of the adversary who picks the elements in the table.
● It is an average over the choices of the algorithm who picks the the hash function from the hash family.

Correct

In order to conclude an expected runtime of $O(1)$ for hash table operations, we assumed the following two happen in some specific order:

- The adversary picks elements $x_1, \ldots, x_n$ for the hash table.
- The algorithm picks a hash function from the hash family.

In what order do these happen?

○ Algorithm first, and then adversary.
● Adversary first, and then algorithm.
○ It does not matter.

Correct

2 Bit lengths

Suppose that there is a toy box with N toys in it. You have a label printer that can print arbitrary strings of 0s and 1s. If you produce labels for the N toys in such a way that each toy gets a unique label, what can be said about the longest label's length?

● $\geq \Omega(\log N)$
○ $\leq O(N)$
○ $\geq \Omega(N)$

Correct

As a remark, for any labeling scheme, the same lower bound of $\Omega(\log N)$ applies even to the average label length, not just the longest label length.

If you produce labels in a way that minimizes the longest label's length, what is this minimum?

○ $\Theta(N)$
○ $\Theta(1)$
● $\Theta(\log N)$

Correct

If our toy box consists of all functions from $\{0, \ldots, M-1\}$ to $\{0, \ldots, n-1\}$, what is the minimum longest label's length?

○ $\Theta(Mn)$
○ $\Theta(M)$
● $\Theta(M \log n)$
○ $\Theta(n \log M)$

Correct

3 Modular arithmetic

Suppose that $M > 1000$ (the universe size) is a prime number. If we pick $a \in \{1, \ldots, M-1\}$ and $b \in \{0, \ldots, M-1\}$, independently and uniformly at random, what is

$$\mathbb{P}_{a,b}[a \times 12 + b = 34 \pmod{M} \text{ and } a \times 56 + b = 78 \pmod{M}]?$$

● $\frac{1}{M(M-1)}$
○ $\frac{1}{M}$
○ $\frac{1}{M^2}$
○ 0

Correct

How about

$$\mathbb{P}_{a,b}[a \times 12 + b = 34 \pmod{M} \text{ and } a \times 56 + b = 34 \pmod{M}]?$$

○ $\frac{1}{M(M-1)}$
○ $\frac{1}{M}$
○ $\frac{1}{M^2}$
● 0

Correct

In fact for any pair of distinct elements x, y in the universe, $u = a \times x + b \mod M$ and $v = a \times y + b \mod M$ are uniformly distributed amongst all distinct pairs.

How many elements of $\{0, \ldots, M-1\}$ are equal to 0 modulo n?

● $\lceil M/n \rceil$
○ $\lfloor M/n \rfloor$
○ $\lfloor M/n \rfloor + 1$

Correct

In fact, for any i, the number of elements from $\{0, \ldots, M-1\}$ equal to i modulo n is $\leq \lceil M/n \rceil$.

Let $u = 0$; pick v uniformly at random from $\{0, \ldots, M-1\} - \{u\} = \{1, \ldots, M-1\}$. What is the chance that $v = u \pmod{n}$?

● $\frac{\lceil M/n \rceil - 1}{M-1}$
○ $\frac{1}{M-1}$
○ $\frac{n}{M-1}$

Correct

You can verify that the answer above is always $\leq 1/n$. The same answer holds as an upper bound if we changed u from 0 to any other element in $\{0, \ldots, M-1\}$.

4 Hash family size

Suppose that we have a universe of size M, and our hash table size is n. If $n \geq M$, what is the minimum size of a universal hash family?

1

Correct

Suppose now that $M = n^{100}$ and we have a nonempty hash family H. Let h^* be one of the hash functions in H. Since $M > n$, h^* must map at least two distinct elements x^*, y^* in the universe to the same bucket (by the pigeonhole principle). What can be said about

$$\mathbb{P}_{h \sim H}[h(x^*) = h(y^*)]?$$

○ $= 0$
● $\geq 1/|H|$
○ $\leq 1/n$

Correct

This means that if H is universal, then

$$1/n \geq \mathbb{P}_{h \sim H}[h(x^*) = h(y^*)] \geq 1/|H|,$$

or in other words $|H| \geq n$. What can be said about the minimum longest 0/1 label length for labeling this hash family?

○ $\geq \Omega(\log n)$
○ $\geq \Omega(\log M)$
● Both of the above.

Correct

For the universal hash family from lecture, how many bits do we need to label the hash functions, if we minimize the longest label's length?

○ $\Theta(M)$
● $\Theta(\log M)$
○ $\Theta(1)$

Correct

This shows the hash family from lecture can be labeled by the optimal number of bits ($\Theta(\log M) = \Theta(\log n)$) when $M = n^{100}$.