1 Hash tables
Hash tables with external 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 hash table.
- It is an average over the choices of the algorithm who picks the hash function from the hash family.

In order to calculate an expected runtime of $O(1)$ for hash table operations, we assumed the following two happenings 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, then adversary.
- Adversary first, and then algorithm.

It does not matter.

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 $k$ bits. 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?

- $O(\log N)$
- $\Theta(k)$
- $\Theta(\log N)$
- $\Omega(k)$

As a remark, for any labeling scheme, the same lower bound of $\Omega(k)$ 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 the minimum?

- $O(\log M)$
- $\Theta(k)$
- $\Theta(\log M)$
- $\Omega(k)$

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

- $\Theta(\log M)$
- $\Theta(\log n)$
- $\Theta(\log N)$
- $\Theta(\log M \cdot \log n)$

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

- Correct
- Incorrect

How about $\Pr(h(a) = h(b))$ and $a, b \in \{0, \ldots, M-1\}$?

- Correct
- Incorrect

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

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

- $\ell M / \ell$
- $\ell M / \ell$
- $\ell M / \ell$
- $\ell M / \ell$

In fact, for any $i$, the $i$th element of $\{0, \ldots, M-1\}$ equal to $i$ modulo $\ell$ is $\ell M / \ell$-th.

Let $a, b \in \mathbb{Z}_M$ be uniformly at random from $\{0, \ldots, M-1\}$ and $a \not\equiv b \pmod{M}$. What is the minimum $n$ such that $a \not\equiv b \pmod{n}$?

- $\Theta(1)$
- $\Theta(\log \log M)$
- $\Theta(\log \log \log M)$
- $\Theta(\log \log \log \log M)$

You can verify that the answer above is always $\Theta(1)$. The same answer holds as an upper bound if we changed $a$ to $a$ modulo $n$ 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 $M \geq n$, what is the minimum size of a universal hash family?

- $\Theta(n)$
- $\Theta(\log n)$
- $\Theta(n \log n)$
- $\Theta(n \log \log n)$

Suppose now that $M = 1000$ 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 $\Pr(h(x) = h(y))$?

- $\Theta(1)$
- $\Theta(\log M)$
- $\Theta(\log \log M)$
- $\Theta(\log \log \log M)$

This means that if $H$ is a universal, then

$$\frac{1}{n} \Pr(h(x) = h(y)) \geq \frac{1}{3}$$

or in other words $|H| \geq n$. What can be said about the minimum largest (or) label length for labeling the hash family?

- $\Theta(\log M)$
- $\Theta(\log (nM))$
- Both of the above.
- Neither of the above.

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

- $\Theta(\log M)$
- $\Theta(\log (nM))$
- $\Theta(\log \log M)$
- $\Theta(\log \log \log M)$

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

Correct