Basic Hashing

dictionaries revisited

Def: *Universe* from which to choose keys

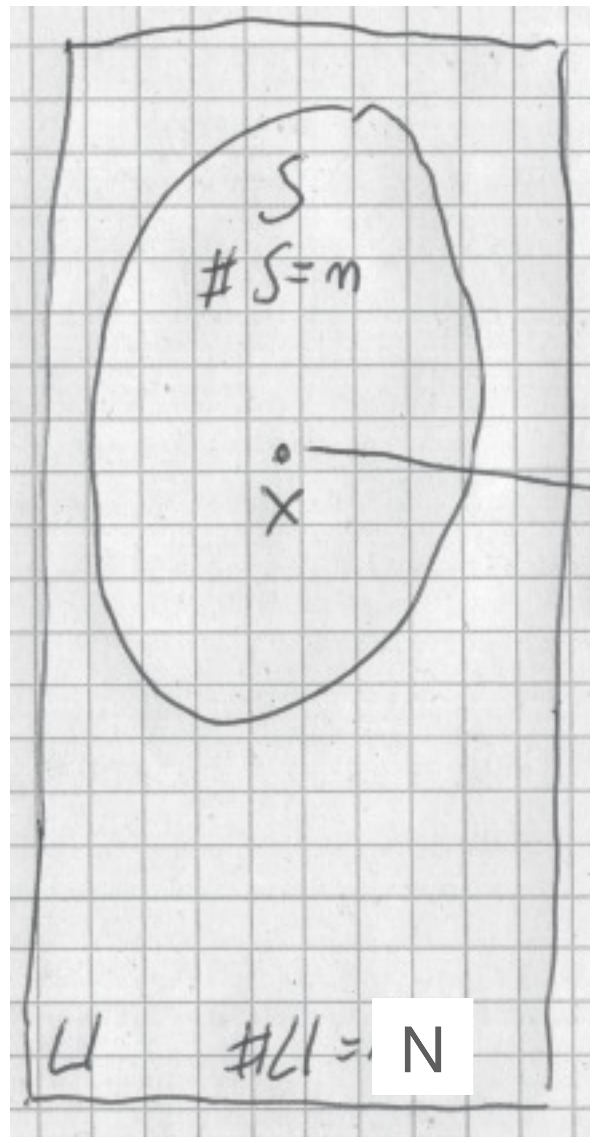
$$U = [0 : N - 1]$$

dictionary for keys x from a smaller subset S of U

$$S \subseteq U \ , \ \#S = n < N$$

dictionary maintains subset $Y \subseteq S$. Operatons for $x \in S$

- $insert(x)$
$$Y' = Y \cup \{x\}$$
- $find(x)$:
$$find(x) = \begin{cases} 1 & x \in Y \\ 0 & x \notin Y \end{cases}$$
- $delete(x)$:
$$Y' = Y \setminus \{x\}$$



dictionaries revisited

Def: *Universe* from which to choose keys

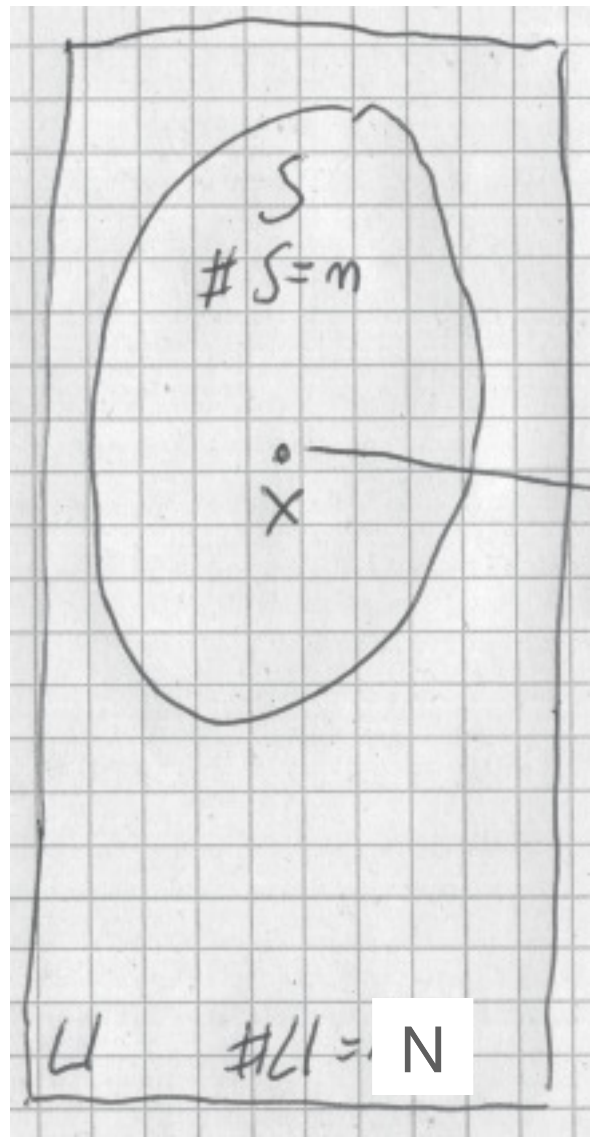
$$U = [0 : N-1]$$

dictionary for keys x from a smaller subset S of U

$$S \subseteq U \ , \ \#S = n < N$$

dictionary maintains subset $Y \subseteq S$. Operatons for $x \in S$

- $insert(x)$
$$Y' = Y \cup \{x\}$$
- $find(x)$:
$$find(x) = \begin{cases} 1 & x \in Y \\ 0 & x \notin Y \end{cases}$$
- $delete(x)$:
$$Y' = Y \setminus \{x\}$$



With balanced trees: time per operation

$$T = O(\log n)$$

basic hashing

- Use hash function

$$h : U \to [0 : m-1]$$

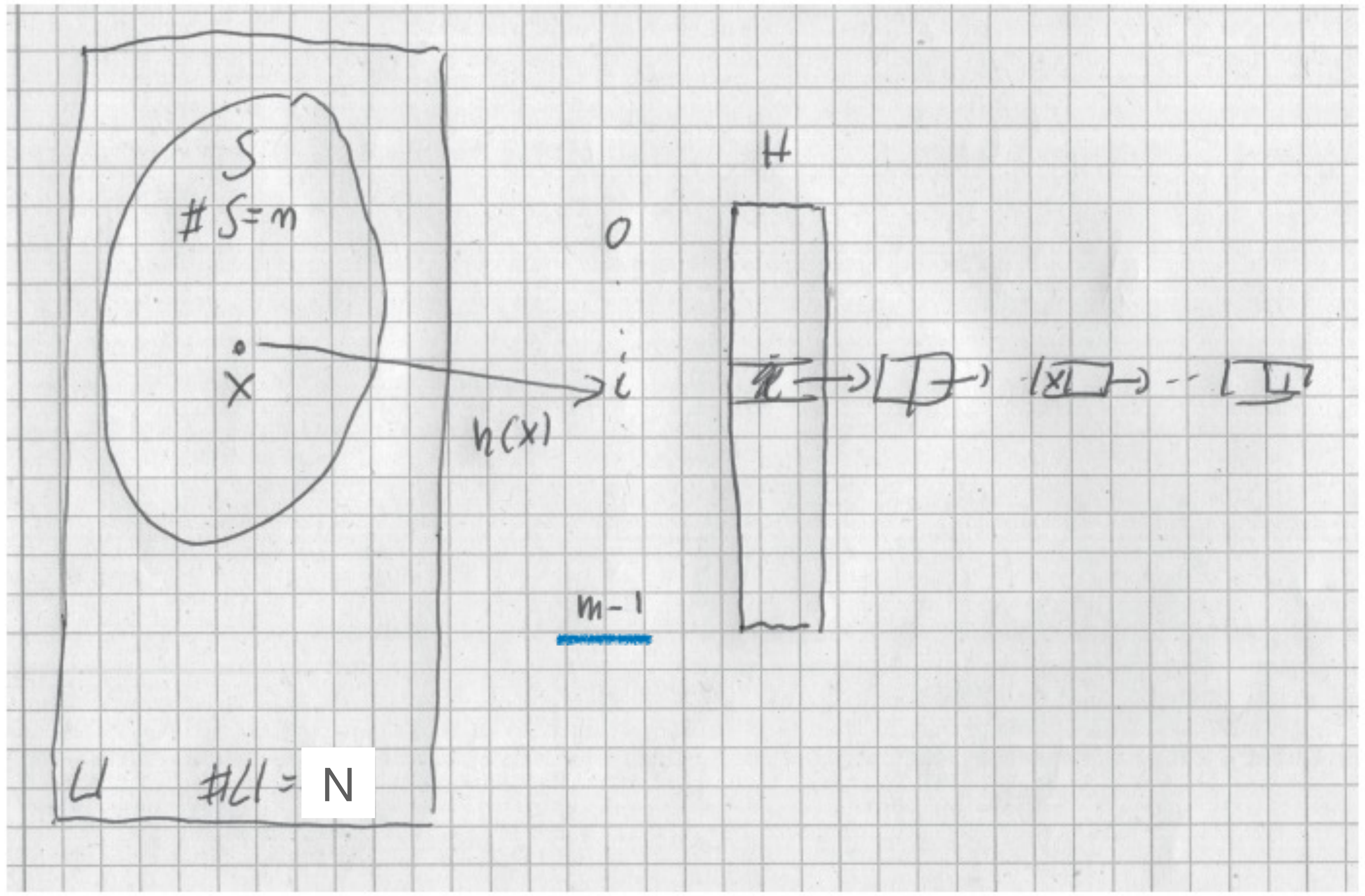


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

basic hashing

- Use hash function

$$h : U \to [0 : m-1]$$

- map keys $x \in U$ to

$$h(x) \in [0 : m-1]$$

  computable in time

$$T = O(1)$$

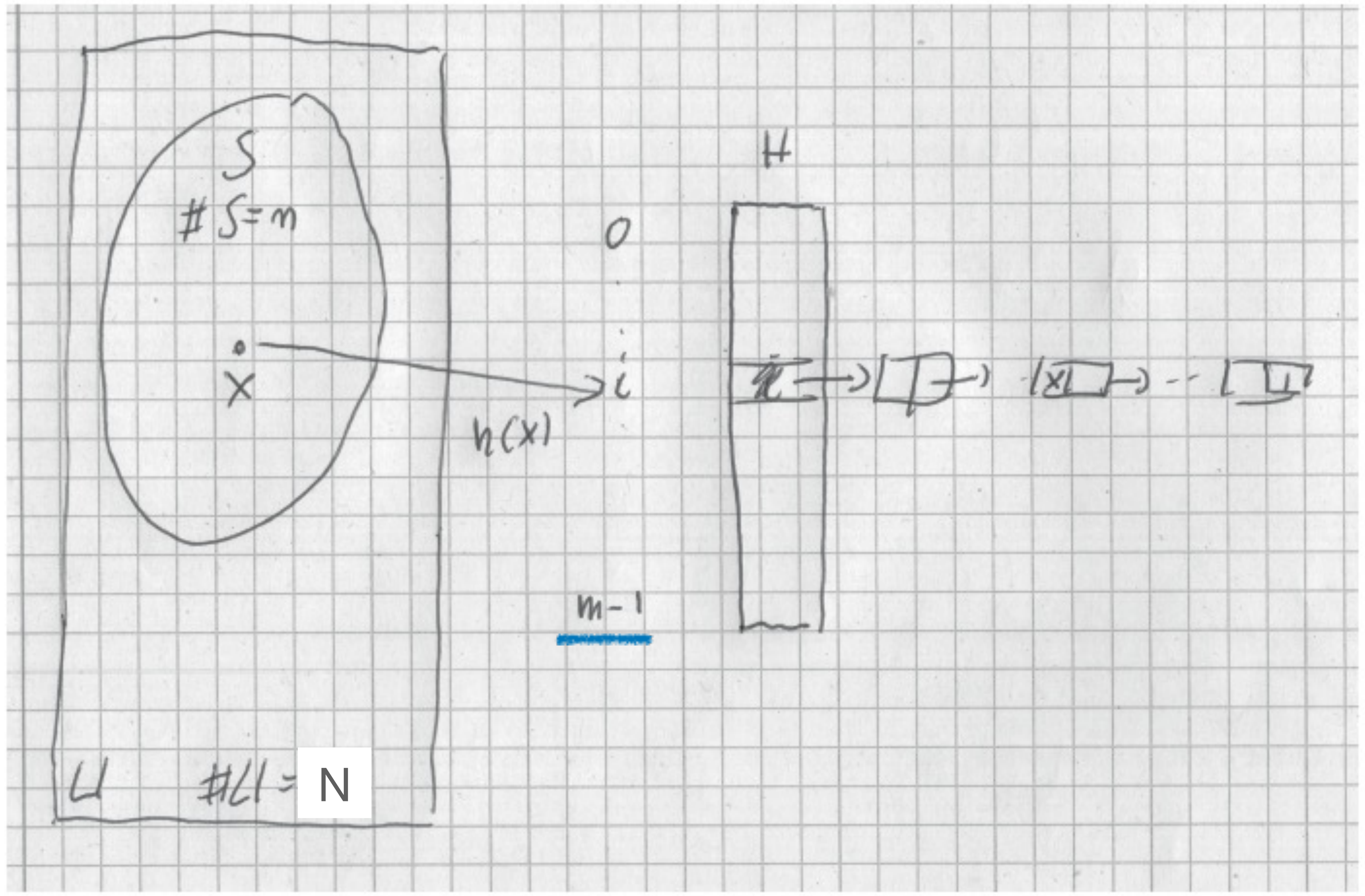


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

basic hashing

- Use hash function

$$h : U \to [0 : m-1]$$

- map keys $x \in U$ to

$$h(x) \in [0 : m-1]$$

- For each hash value

$$i \in [0 : m-1]$$

maintain linked list $L(i)$ of elements in Y which are hashed to i. Abusing notation

$$x \in L(i) \leftrightarrow x \in Y \ \wedge \ h(x) = i$$

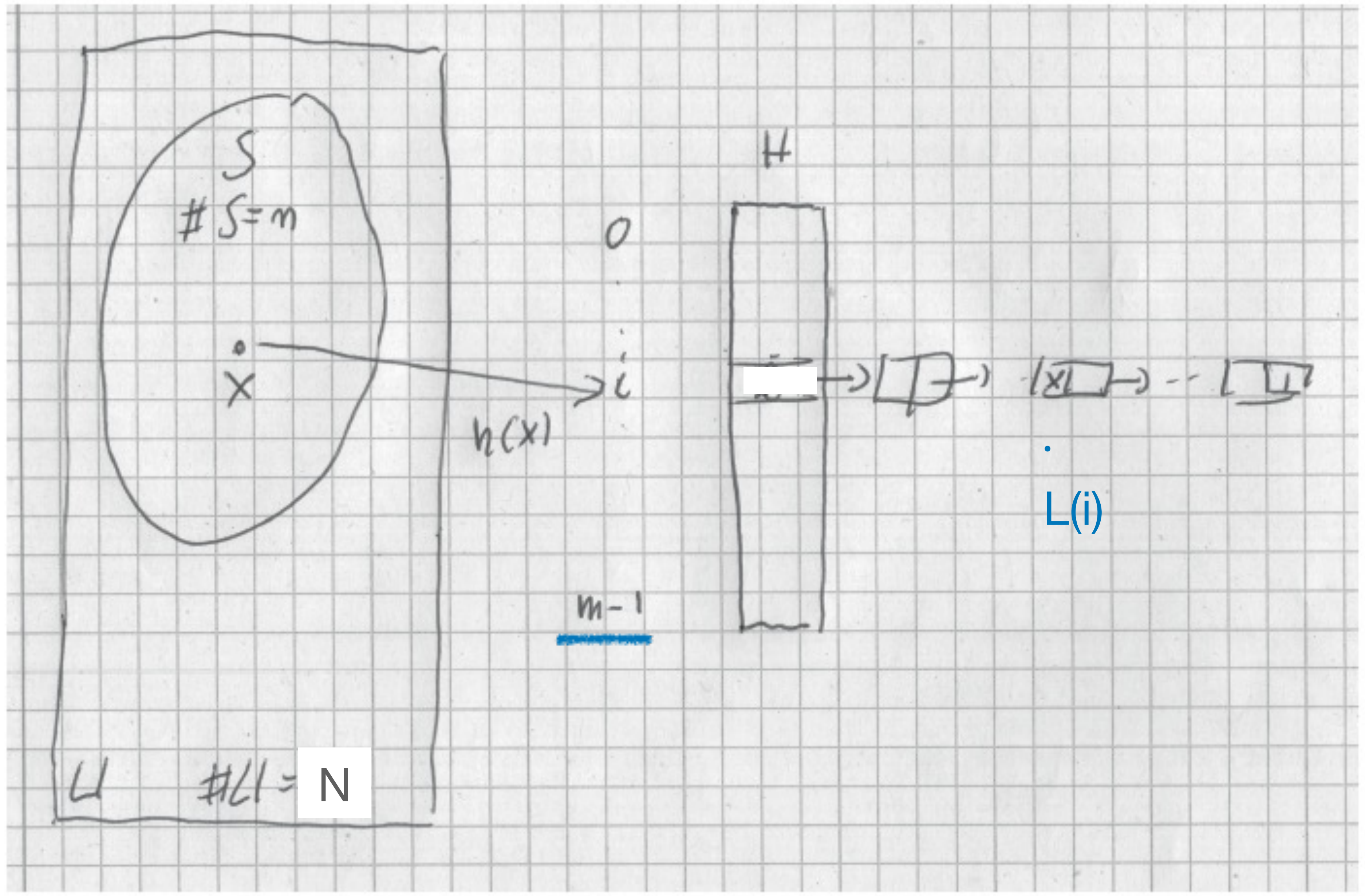


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

basic hashing

- Use hash function

$$h : U \to [0 : m-1]$$

- map keys $x \in U$ to

$$h(x) \in [0 : m-1]$$

- For each hash value

$$i \in [0 : m-1]$$

maintain linked list $L(i)$ of elements in Y which are hashed to i. Abusing notation

$$x \in L(i) \leftrightarrow x \in Y \ \wedge \ h(x) = i$$

- in an array H of length m maintain pointers to the heads of the lists

$$H[i]* = hd(L(i))$$

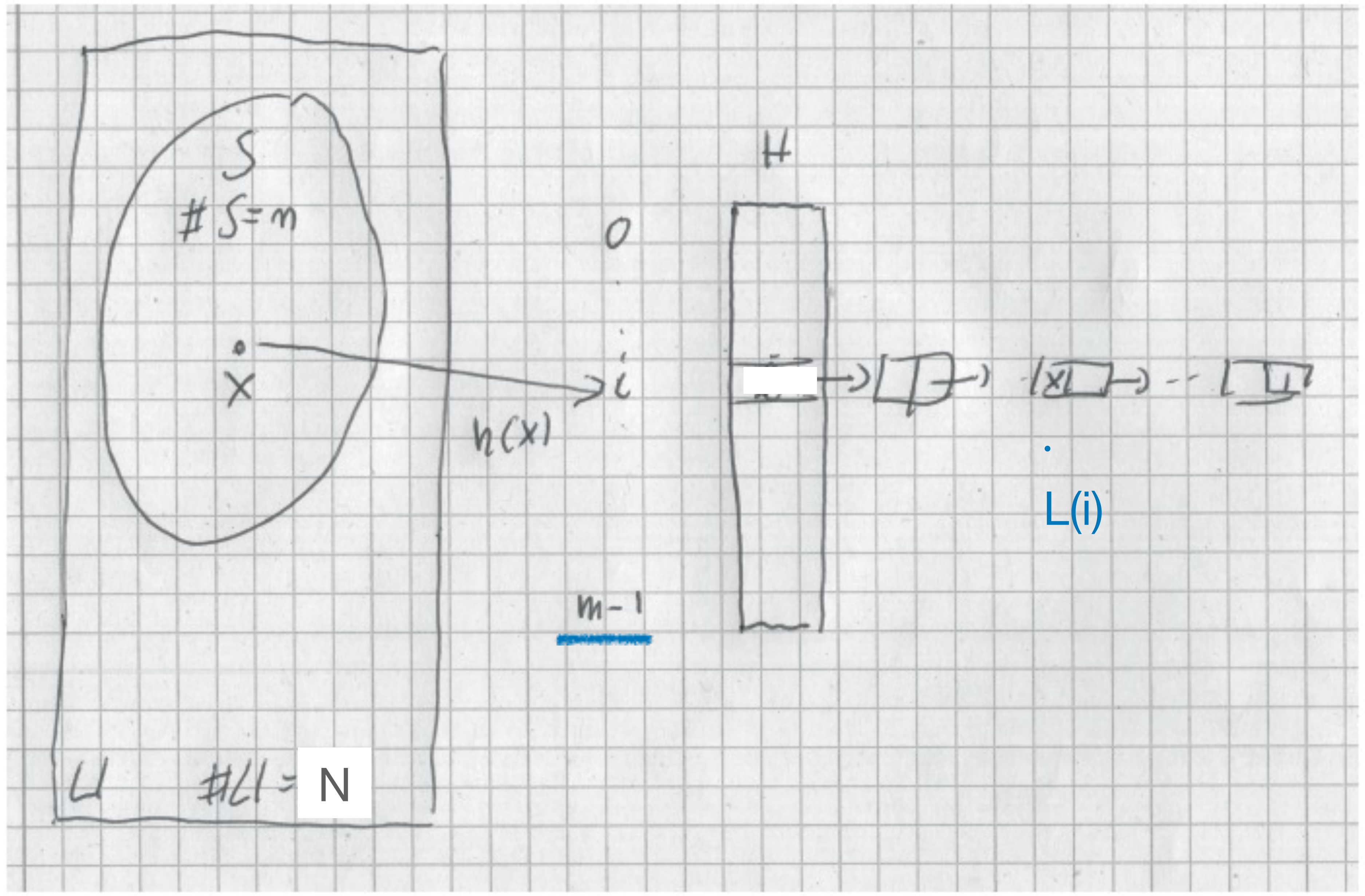


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

distributing keys evenly

def: hash function h *distributes keys evenly* if

$$\forall i.\ \#\{x \in U \mid h(x) = i\} \leq \lceil N/m \rceil$$

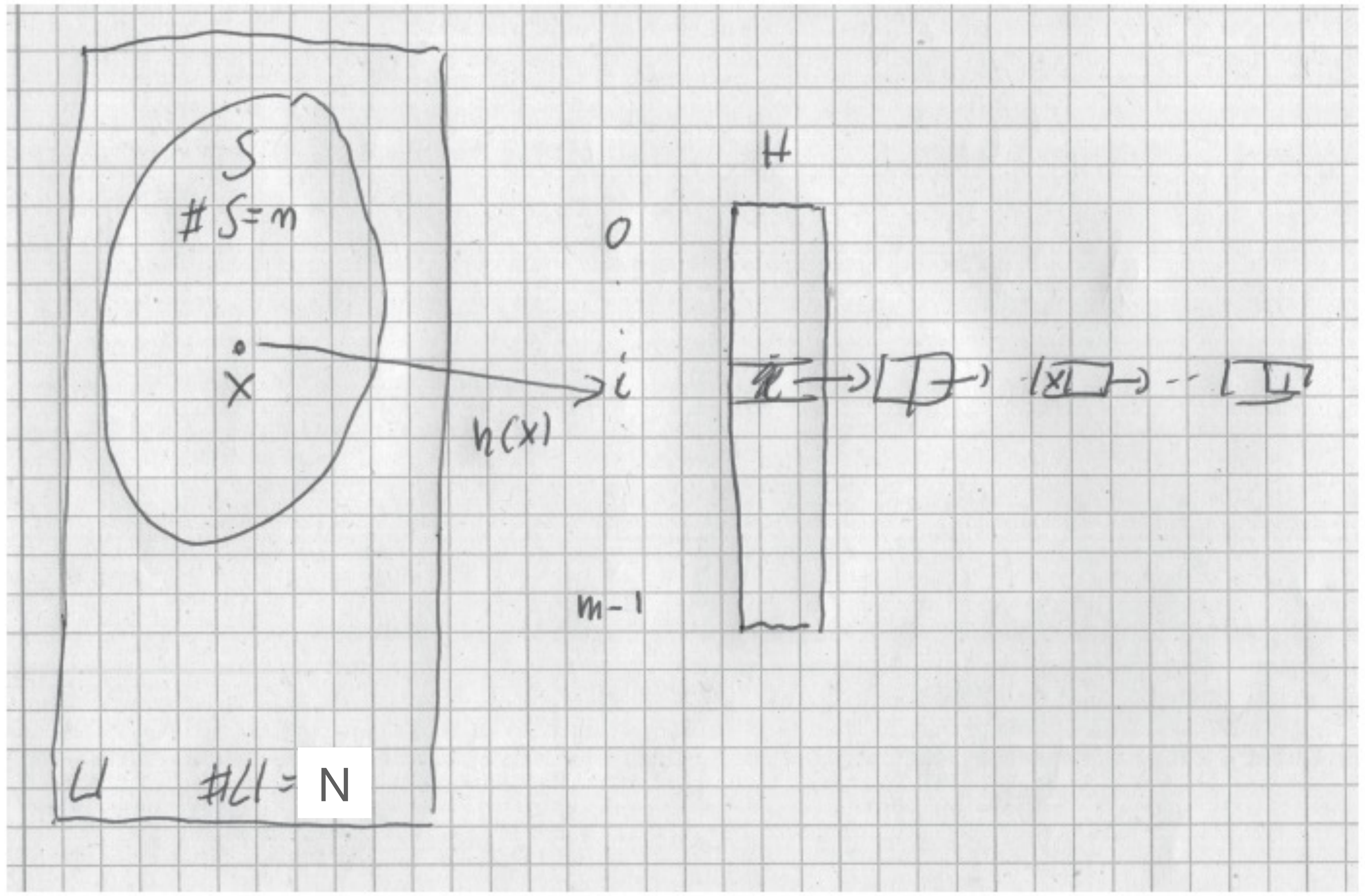


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

distributing keys evenly

def: hash function h *distributes keys evenly* if

$$\forall i.\ \#\{x \in U \mid h(x) = i\} \le \lceil N/m \rceil$$

Example:

$$N = 2^{\nu}\ ,\ m = 2^{\mu}$$

- represent $x = x[\nu - 1 : 0]$ as binary number of length ν
$$bin(x) \in \mathbb{B}^{\nu}$$
- pick subset of μ indices in $[0 : \nu - 1]$
$$0 \le j_0 < \ldots < j_{\mu-1} < \nu$$
- concatenate bits $bin(x)[j_y]$
$$z = bin(x)[i_{\mu-1}], \ldots, bin(x)[0]$$
and interpret as number
$$h(x) = \langle z \rangle$$

Even distribution of keys: exercise.

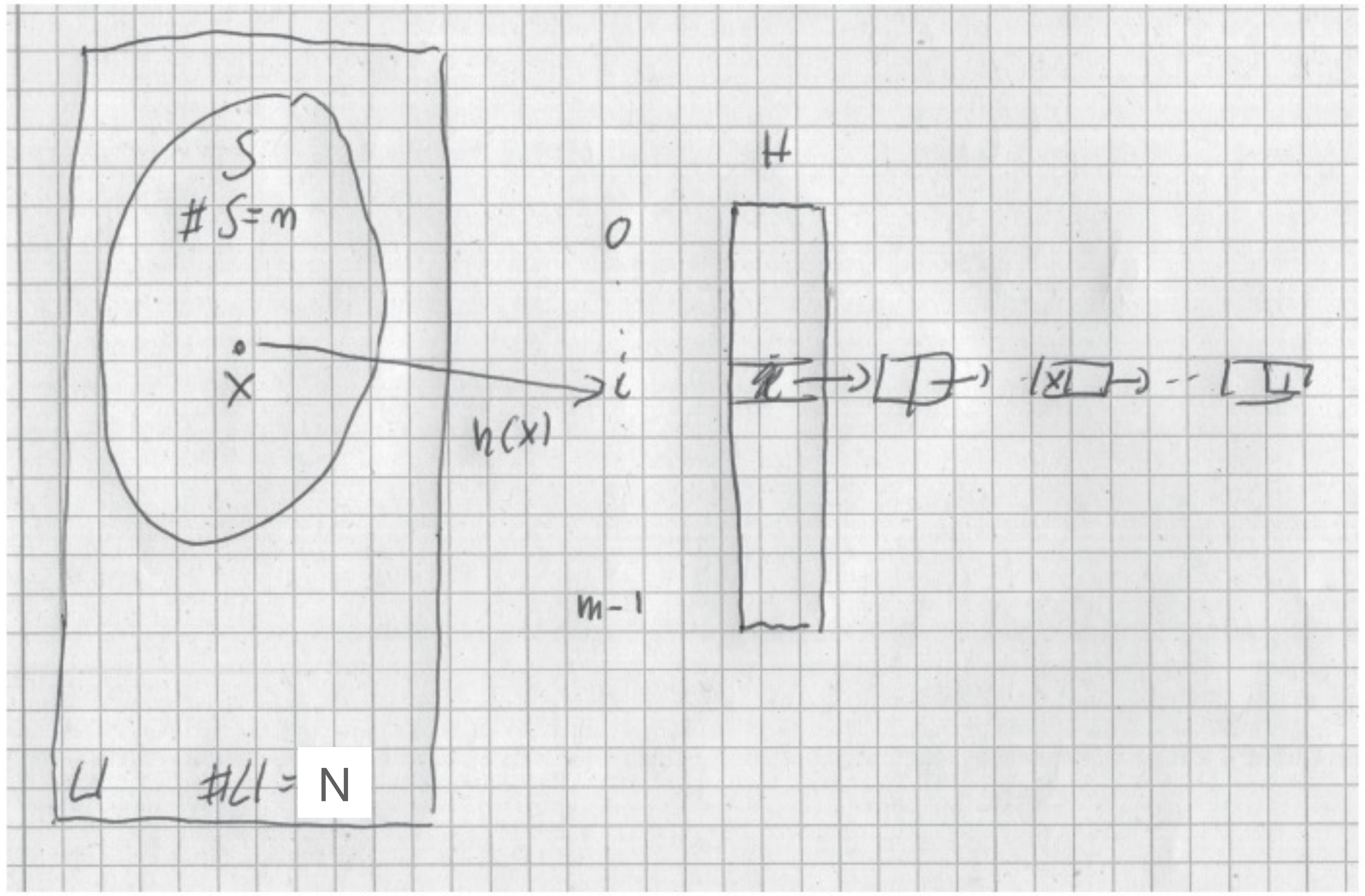


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

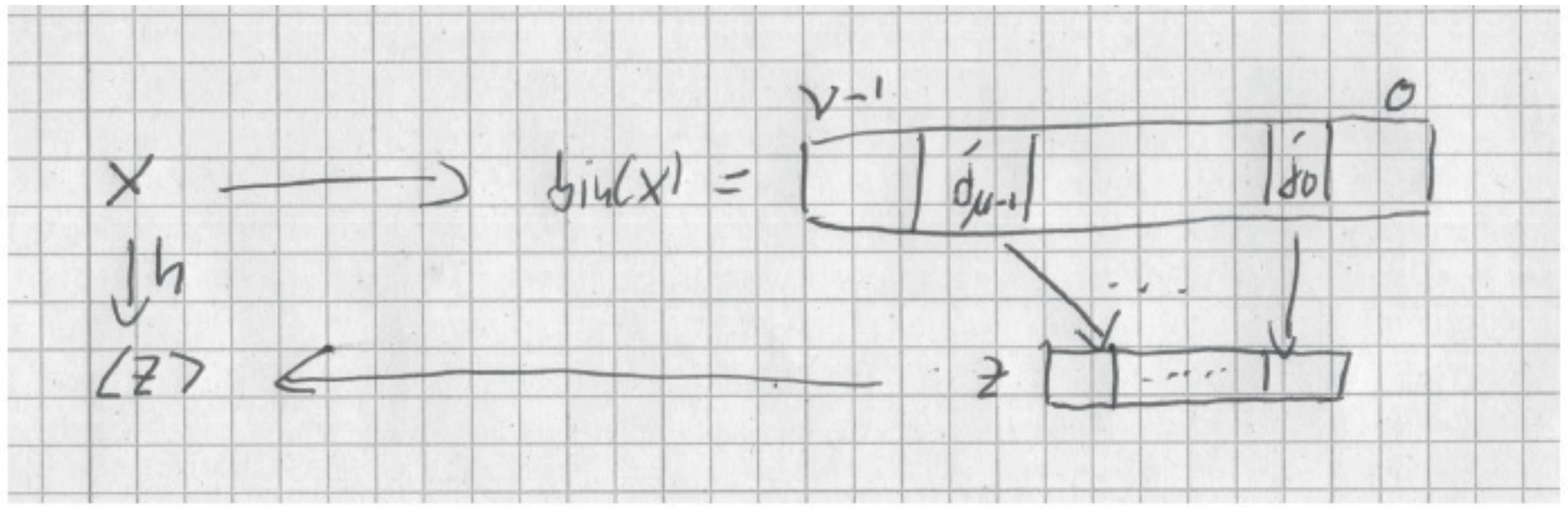


Figure 2: To hash x convert it into a binary number $bin(x)$ of lenght ν, obtain z by concatenating a prescribed subsequence of μ bits of $bin(x)$ and convert back

distributing keys evenly

def: hash function h *distributes keys evenly* if

$$\forall i.\ \#\{x \in U \mid h(x) = i\} \leq \lceil N/m \rceil$$

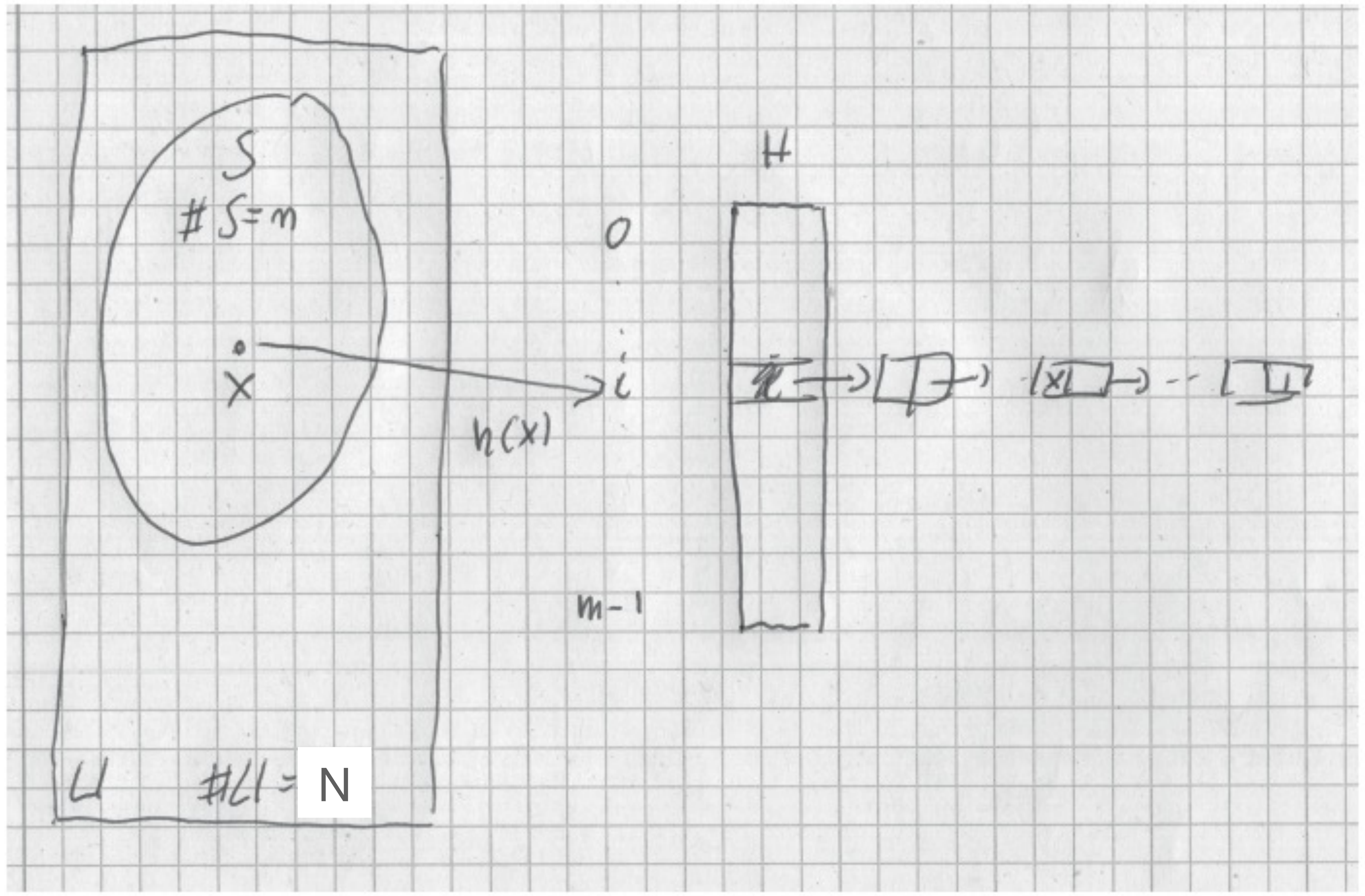


Figure 1: A key $x \in S$ is stored as an element of list $L(i)$, where $i = h(x)$ is the hash value of x. Pointers to the heads of lists $L(i)$ are stored in an array $H(i)$. Run time of list operations on x is bounded by $O(|L(i)|)$

hope: if h distributes keys evenly *and* each set S is occurs randomly with probability :

$$p(S) = \binom{N}{n}^{-1}$$

then for *load factor*

$$\alpha = n/m$$

the expected length of lists $L(i)$ (and hence the expected run time of operations) is

$$E(|L(i)|) = O(\alpha) = O(n/m)$$

distribution of sets S

usually missing in textbook or lecture notes

recall: *random permutations*

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

recall: *random permutations*

distribution of sets S

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

recall: *random permutations*

distribution of sets S

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

recall: *random permutations*

distribution of sets S

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

proof:

$$\begin{aligned} p\{S(\pi) = S\} &= \frac{\#\{\pi \mid S(\pi) = S\}}{N!} \\ &= \frac{n! \cdot (N-n)!}{N!} \\ &= \binom{N}{n}^{-1} \end{aligned}$$

recall: *random permutations*

distribution of sets S

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \rightarrow [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p) \ , \ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

proof:

$$\begin{aligned} p\{S(\pi) = S\} &= \frac{\#\{\pi \mid S(\pi) = S\}}{N!} \\ &= \frac{n! \cdot (N-n)!}{N!} \\ &= \binom{N}{n}^{-1} \end{aligned}$$

Lemma 2. *For all $k \in [0:n-1]$ and $y \in U$ the probability that the k'th chosen element $\pi(k)$ is y equals*

$$p\{\pi(k) = y\} = 1/N$$

distribution of sets S

recall: *random permutations*

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

proof:

$$\begin{aligned} p\{S(\pi) = S\} &= \frac{\#\{\pi \mid S(\pi) = S\}}{N!} \\ &= \frac{n! \cdot (N-n)!}{N!} \\ &= \binom{N}{n}^{-1} \end{aligned}$$

Lemma 2. *For all $k \in [0:n-1]$ and $y \in U$ the probability that the k'th chosen element $\pi(k)$ is y equals*

$$p\{\pi(k) = y\} = 1/N$$

proof:

$$\begin{aligned} p\{\pi(k) = y\} &= \frac{\#\{\pi \mid \pi(k) = y\}}{N!} \\ &= \frac{(N-1)!}{N!} \\ &= 1/N \end{aligned}$$

distribution of sets S

recall: *random permutations*

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

proof:

$$\begin{aligned} p\{S(\pi) = S\} &= \frac{\#\{\pi \mid S(\pi) = S\}}{N!} \\ &= \frac{n! \cdot (N-n)!}{N!} \\ &= \binom{N}{n}^{-1} \end{aligned}$$

Lemma 2. *For all $k \in [0:n-1]$ and $y \in U$ the probability that the k'th chosen element $\pi(k)$ is y equals*

$$p\{\pi(k) = y\} = 1/N$$

proof:

$$\begin{aligned} p\{\pi(k) = y\} &= \frac{\#\{\pi \mid \pi(k) = y\}}{N!} \\ &= \frac{(N-1)!}{N!} \\ &= 1/N \end{aligned}$$

Lemma 3. *If hash function h distributes keys evely, then for all $k \in [0:n-1]$ and $i \in [0:m-1]$ the probability that the k'th chosen element $\pi(k)$ is mapped to i is bounded as*

$$p\{h(\text{[redacted]}(\pi(k))) = i\} \leq 1/m + 1/N$$

distribution of sets S

recall: *random permutations*

Consider permutations of U

$$\Pi_N = \{\pi \mid \pi : [0:N-1] \to [0:N-1] \quad \text{bijective}\}$$

Pick permutation π random and equally distributed from Π_N. Probability space

$$W = (\Pi_N, p)\ ,\ p(\pi) = \frac{1}{N!} \text{ for all } \pi$$

see slide set 'hiring problem'

Form random set S with first n elements of π

$$\boxed{S(\pi) = \{\pi(j) \mid j < n\}}$$

Lemma 1. *All subsets of U with k elements are chosen with equal probability: for all $S \subseteq U$ with $\#S = n$*

$$p\{S(\pi) = S\} = \binom{N}{n}^{-1}$$

proof:

$$\begin{aligned} p\{S(\pi) = S\} &= \frac{\#\{\pi \mid S(\pi) = S\}}{N!} \\ &= \frac{n! \cdot (N-n)!}{N!} \\ &= \binom{N}{n}^{-1} \end{aligned}$$

Lemma 2. *For all $k \in [0:n-1]$ and $y \in U$ the probability that the k'th chosen element $\pi(k)$ is y equals*

$$p\{\pi(k) = y\} = 1/N$$

proof:

$$\begin{aligned} p\{\pi(k) = y\} &= \frac{\#\{\pi \mid \pi(k) = y\}}{N!} \\ &= \frac{(N-1)!}{N!} \\ &= 1/N \end{aligned}$$

Lemma 3. *If hash function h distributes keys evely, then for all $k \in [0:n-1]$ and $i \in [0:m-1]$ the probability that the k'th chosen element $\pi(k)$ is mapped to i is bounded as*

$$p\{h(\pi(k)) = i\} \leq 1/m + 1/N$$

proof:

$$\begin{aligned} p\{h(\pi(k)) = i\} &= \sum_{h(y)=i} p\{\pi(k) = y\} \\ &= \sum_{h(y)=i} (1/N) \quad \text{(lemma 2)} \\ &\leq \lceil N/m \rceil \cdot (1/N) \\ &\leq (N/m + 1) \cdot (1/N) \end{aligned}$$

expected length of list $L(i)$

hope: if h distributes keys evenly *and* each set S is occurs randomly with probability :

$$p(S) = \binom{N}{n}^{-1}$$

then for *load factor*

$$\alpha = n/m$$

the expected length of lists $L(i)$ (and hence the expected run time of operations) is

$$E(|L(i)|) = O(\alpha) = O(n/m)$$

def: indicator variable

$$X_{k,i} = \begin{cases} 1 & h(\pi(k)) = i \\ 0 & \text{otherwise} \end{cases}$$

expected length of list $L(i)$

hope: if h distributes keys evenly *and* each set S is occurs randomly with probability :

$$p(S) = \binom{N}{n}^{-1}$$

then for *load factor*

$$\alpha = n/m$$

the expected length of lists $L(i)$ (and hence the expected run time of operations) is

$$E(|L(i)|) = O(\alpha) = O(n/m)$$

def: indicator variable

$$X_{k,i} = \begin{cases} 1 & h(\pi(k)) = i \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned} E(|L(i)|) &\leq E(\sum_{k=0}^{n-1} X_{k,i}) \\ &= \sum_{k=0}^{n-1} E(X_{k,i}) \quad \text{(linearity)} \\ &= \sum_{k=0}^{n-1} p\{h(\pi(k)) = i\} \quad \text{(indicator variable)} \\ &\leq n \cdot (1/m + 1/N) \quad \text{(lemma 3)} \\ &= n/m + 1 \quad (n \leq N) \end{aligned}$$

expected length of list $L(i)$

Lemma: if h distributes keys evenly *and* each set S is occurs randomly with probability :

$$p(S) = \binom{N}{n}^{-1}$$

then for *load factor*

$$\alpha = n/m$$

the expected length of lists $L(i)$ (and hence the expected run time of operations) is

$$E(|L(i)|) = O(\alpha) = O(n/m)$$

def: indicator variable

$$X_{k,i} = \begin{cases} 1 & h(\pi(k)) = i \\ 0 & \text{otherwise} \end{cases}$$

$$\begin{aligned}
E(|L(i)|) &\leq E(\sum_{k=0}^{n-1} X_{k,i}) \\
&= \sum_{k=0}^{n-1} E(X_{k,i}) \quad \text{(linearity)} \\
&= \sum_{k=0}^{n-1} p\{h(\pi(k)) = i\} \quad \text{(indicator variable)} \\
&\leq n \cdot (1/m + 1/N) \quad \text{(lemma 3)} \\
&= n/m + 1 \quad (n \leq N)
\end{aligned}$$