Lecture 8

Hash Tables with Linear Probing

We saw hashing with chaining. Using universal hashing we get $O(1)$ expected time per operation.

One disadvantage is that chaining requires a list data structure at each bucket.

Today we will discuss another popular technique called linear probing. Mostly following Kent Quanrud's notes for this. He has nice figures and more detailed explanation, including historical notes. For this reason we will be high-level in our description.

Linear Probing

$\mathcal{U}$ universe, $|\mathcal{U}| = N$

k size of hash table $A[0..m-1]$

Pick random hash function h from a hash family $\mathcal{H}$.

insert(x)

- $i = h(x)$
- while (A[i] is not empty)

    $i = i+1 \mod k$

- $A[i] = x$.

find (x)

- $i = h(x)$
- While $A[i] \neq$ empty do
  if $A[i] = x$ output Yes

- Output No.

delete (x) is more complicated

different strategies but we want to maintain insert and find correctness. So

to delete (x) we first find (x).

Say x is in $A[i]$.
If $h(x) = i$ then we

set $A[i]$ = empty and we are done.

Otherwise we has inserted x into a location "after" $h(x)$ due to collisions. If we remove x from $A[i]$ we create a "hole". We try to fill it by scanning from i to the right to see if we encounter another element y s.t. $A[j] = y$ but $h(y) \neq j$ and move it to the hole. We repeat this.

More formally.

delete (x)

$i \leftarrow find(x)$ (assume x in A)

$A[i] \leftarrow$ empty

Repeat:

$j \leftarrow i+1 \pmod{k}$.

While ($A[j] \neq$ empty and $h(A[j]) = A[j]$)

$j \leftarrow j+1 \mod k$

If $A[j] =$ empty then Break

Else

$A[i] = A[j]$

$i \leftarrow j$

Until (TRUE)

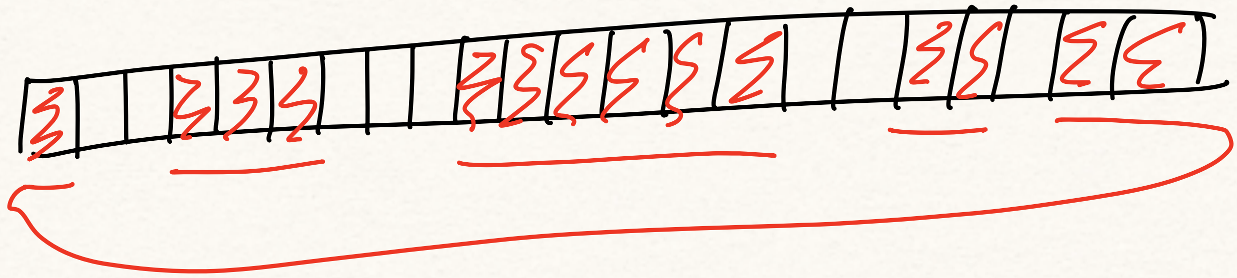
Analysis

Assuming "ideal" hash functions.

How can we upper bound the cost of the operations?

Suppose we do n operations.
Let S be the elements that were ever considered.
Assume $m > 2en$. Then we will consider the state of the hash table as if we had inserted all the elements in S

The hash table A will be broken into "runs" -



where a run is a maximal "interval" of occupied cells. We observe the following. The cost of insert(x), find(x) and delete(x) are proportional / upperbounded by $|R(x)|$ where $R(x)$ is the run that contains $h(x)$.

Thus fixing S to be of size n and fixing an element x we want to know the following.

What is $E[|R(x)|]$?

We will use R for $R(x)$.

R is a random subset of S depending on h.

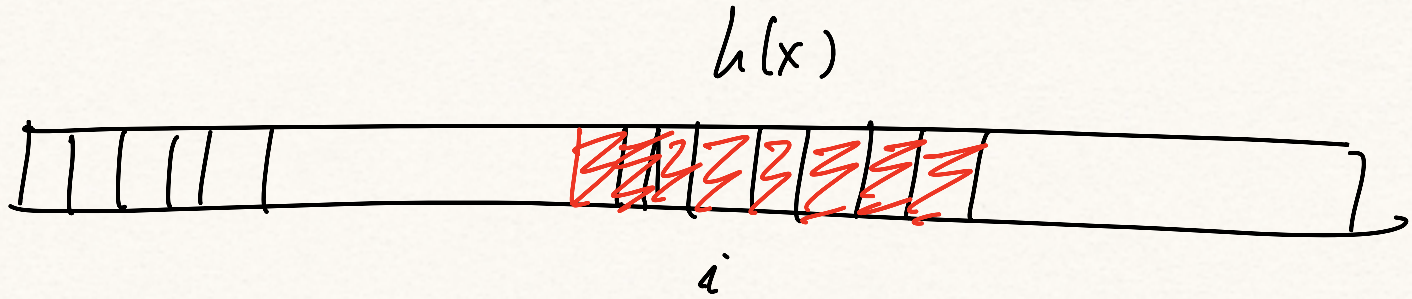
Lemma: If $m > 2en$ then

$$E[R] = O(1).$$

Assuming above we see that expected cost of the n operations is $O(n)$ if $m > 2en$.

Proof of Lemma:

Suppose $h(x) = i$



$$E[|R|] = \sum_{l=1}^{n} l \cdot \Pr[|R| = l].$$

What is $\Pr[|R| = l]$.

Consider an "interval" I that contains i and $|I| = l$.

There are l such intervals. Say $I_1, I_2, \ldots, I_l$.

Thus $\Pr[|R| = l] = l \Pr[R = I_j]$

by symmetry.

What is $\Pr[R = I_j]$?

$|I_j| = l$

exactly l items out of n hash to I_j, and there are empty slots next to I_j but we will ignore the second part

$\Pr[\text{exactly } l \text{ items of } S \text{ hash to } I_j]$

$$\leq \binom{n}{l}\left(\frac{l}{m}\right)^l$$

$$\leq \left(\frac{en}{l}\right)^l \cdot \left(\frac{l}{m}\right)^l$$

$$\leq \left(\frac{en}{m}\right)^{l} \leq \frac{1}{2^{l}}$$

if $m \geq 2en$.

$$\Rightarrow E[|R|] \leq \sum_{l=1}^{n} l \cdot \left[l \cdot \frac{1}{2^{l}} \right]$$

$$\leq \sum_{l=1}^{n} l^{2} \cdot \frac{1}{2^{l}} = O(1).$$

$\square$

The above analysis assumed ideal hashing. Can we obtain a similar result with "normal" hashing? It turns out that it suffices to assume 5-universal hash functions.

<u>Lemma</u>: Suppose $h \sim H$ where H is 5-strongly universal family from $[U] \to [m]$ and $m \geq 8n$. Then expected cost of each of the first n operations is $O(1)$.

In order to analyze this we need a concentration lemma for 4-wise independent random variables which generalizes Chebyshev's inequality.

<u>Lemma</u>: Suppose $X_1, X_2, \ldots, X_n \in \{0,1\}$ and are 4-wise independent.
Let $X = \sum X_i$

Let $\mu = E[X]$. Then

$$\Pr[X \geq \mu + \beta] \leq \frac{\mu + 3\mu^2}{\beta^4}.$$

Lets assume Lemma and prove the bound on $E[|R|]$.

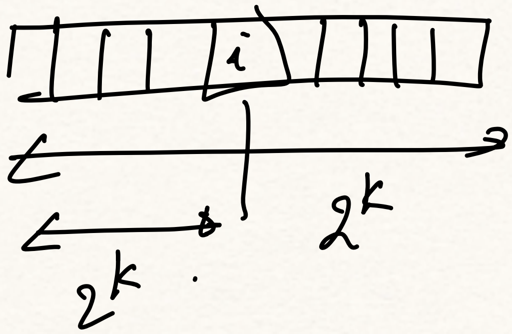
$$E[|R|] \leq \sum_{l=1}^{n} l \cdot \Pr[|R| = l]$$

$$\leq \sum_{k=1}^{\lceil \log n \rceil} 2^k \Pr[2^{k-1} < |R| \leq 2^k].$$

We now upper bound $\Pr[2^{k-1} < |R| \leq 2^k]$.

Let $h(x) = i$ and consider "interval" I_k with length 2^k centered at i

I_k.



If $2^{k-1} < |R| \leq 2^k$ then the event A happens where A is 2^{k-1} items from S hash into I_k.

So we will use $\Pr[A]$ as upper bound.

Let $X = \sum_{a \in S} X_a$

where X_a is indicator of $a \in S$ hashing into I_k conditioned on $h(x) = i$. Since $\mathcal{H}$ was 5-strongly universal, X_a $a \in S$ are 4-wise independent:

$$E[X] = \sum_{a \in S} X_a = \sum_{a \in S} \Pr[a \in I_k]$$

$$\leq n \cdot \frac{|I_k|}{m}$$

$$\leq \frac{n \cdot 2^{k+1}}{8n} \leq 2^{k-2}$$

$$\Pr[A] = \Pr[X \geq 2^{k-1}]$$

$$= \Pr[X \geq E[X] + 2^{k-2}]$$

$$\leq \frac{4 \cdot (2^{k-2})^2}{(2^{k-2})^4}$$

$$\leq 4 \cdot \frac{1}{(2^{k-2})^2}.$$

Now

$$E[|R|] \leq \sum_{k=1}^{\lceil \ln n \rceil} 2^k \Pr\left[2^{k-1} < |R| \leq 2^k\right]$$

$$\leq \sum_{k=1}^{\lceil \ln n \rceil} 2^k \cdot 4 \cdot \frac{1}{2^{2k-4}}$$

$$= O(1).$$

D.

Lemma: Suppose $X_1, X_2, \ldots, X_n \in \{0,1\}$ and are 4-wise independent.
Let $X = \sum X_i$

Let $\mu = E[X]$. Then

$$\Pr[X \geq \mu + \beta] \leq \frac{\mu + 3\mu^2}{\beta^4}.$$

Proof:

$$\Pr[X - \mu \geq \beta] = \Pr[(X-\mu)^4 \geq \beta^4]$$

$$\leq \frac{E[(X-\mu)^4]}{\beta^4}$$

by Markov.

Need to bound $E[(X-\mu)^4]$ by $\mu+3\mu^2$.

$X-\mu = \sum_i (X_i - p_i)$. Let $Y_i = X_i - p_i$

$E[Y_i]=0$.

$$E[(X-\mu)^2] = E\left[\left(\sum_{i=1}^{n} Y_i\right)^4\right]$$

$$\left(\sum_i Y_i\right)^4 = \sum_{i\in[n]} \sum_{j\in[n]} \sum_{k\in[n]} \sum_{l\in[n]} Y_i Y_j Y_k Y_l.$$

$$E\left[\left(\sum_i Y_i\right)^4\right] = \sum_{i\in[n]} E[Y_i^4] + 6\sum_{i=1}^{n} E[Y_i^2]\sum_{j=i+1}^{n} E[Y_j^2]$$

By 4-wise independence and $E[Y_i]=0$ any term with only one occurrence of Y_i goes to 0.

$$= \sum_{i=1}^{n}\left[p_i(1-p_i)^4 + (1-p_i)p_i^4\right]$$
$$+ \; 6\sum_{i=1}^{n}\left(p_i(1-p_i)^2 + (1-p_i)p_i^2\right)$$

$$\sum_{j=i+1}^{n} \left(p_j (1-p_j)^2 + (1-p_j) p_j^2 \right)$$

$$\leq \sum p_i + 6 \sum_{i=1}^{n} p_i \sum_{j=i+1}^{n} p_j$$

$$\leq \sum p_i + 3 \left(\sum_{i=1}^{n} p_i \right)^2$$

$$\leq \mu + 3\mu^2 .$$

$\square$.