COL 702 Lecture 9 Aug 20

An example universal hash family

Carter-Wegman

$U = \{0, 1, \ldots N-1\}$

$T = \{0, 1, \ldots m-1\}$

$h : U \to T$

$\forall\ x, y \in U \quad \sum_{h \in H} \delta_h(x, y) \leq c \frac{|H|}{m}$

$h_a = \left((x + a) \bmod N\right) \bmod m \qquad a \in U,\ a \neq 0$

$|H| = N - 1$

$x, y : \quad y = x + m . \quad \delta_a(x, y) = 1$ iff

$\left((y + a) \bmod N\right) \bmod m = \left((x + a) \bmod N\right) \bmod m$

For how many a's the above eqn holds?

$\left(\left(y + a\right) \bmod N\right) \bmod m = y \bmod N \bmod m + a \bmod N \bmod m$

$= (x + m) \bmod N \bmod m + a \bmod N \bmod m$

$= x \bmod N \bmod m + m \bmod N \bmod m + \ldots$

$m \bmod N \bmod m = 0 \quad m < N$

$= (x + a) \bmod M \bmod n$ for any a

Not universal

Multiplicative

$h_a = (ax \bmod N) \bmod m$ and suppose N is prime

for any prime p, if we consider the set of elements $\{1, 2, \ldots p-1\}$ then they form a group $\mathbb{Z}^*_p$

for any element $i \in \mathbb{Z}^*_p$, there exists a unique inverse i^{-1}, s.t $i \cdot i^{-1} \equiv_p 1$

Bertrand's postulate: Between any integers K, $2K$, there is a prime no.

$\delta_a(x, y) = 1$ if

$$(ax \bmod N) \bmod m = (ay \bmod N) \bmod m \quad (*)$$

How many solns exist?

$$\Rightarrow \quad ax \bmod N = ay \bmod N + km$$

$$= (ay + km) \bmod N$$

$$\Rightarrow \quad a(x-y) \equiv_N km$$

$$\Rightarrow \quad a \equiv_N (x-y)^{-1} km$$

for $k=1$, there is a unique soln

for any k, there is a unique soln

$$k : \{ \pm 1, \pm 2, \ldots \lceil \tfrac{N}{m} \rceil \}$$

So # collisions $\leq \dfrac{2 \cdot \lceil N \rceil}{m}$

$|H| = N-1$

Is $2 \cdot \lceil \frac{N}{m} \rceil \overset{?}{\leq} c \cdot \dfrac{N-1}{m}$

Note: Hash functions must be easily computable for them to be useful, i.e in $O(1)$-time

Sorting strings

Given any finite alphabet Σ

a string is an ordered set of symbols over Σ

string $\in \Sigma^*$

What is the ordering?

Dictionary / Lexicographic ordering

Interpolation search : $O(\log \log n)$ expected time

Eg

Strings cave, at, bat

at, bat, cave, cavern

prefix

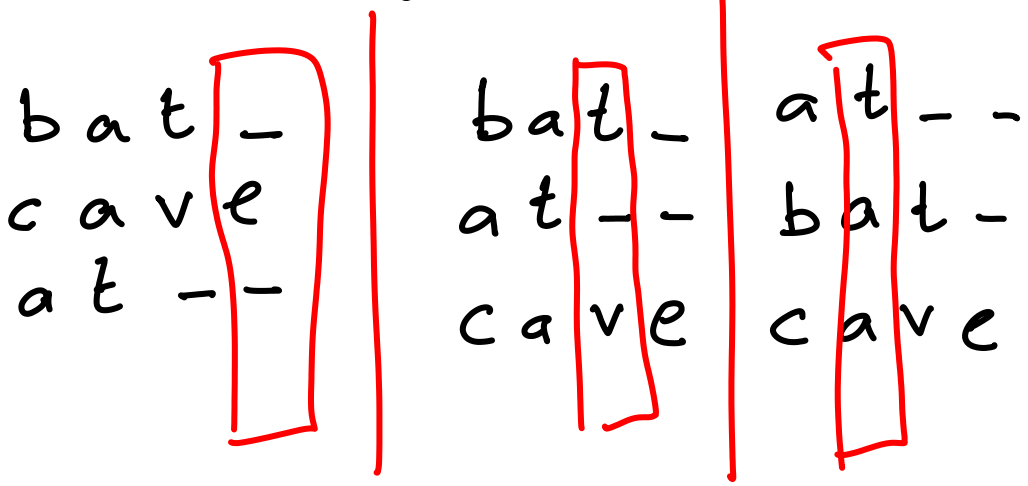
Time to compare : proportional to the length of strings

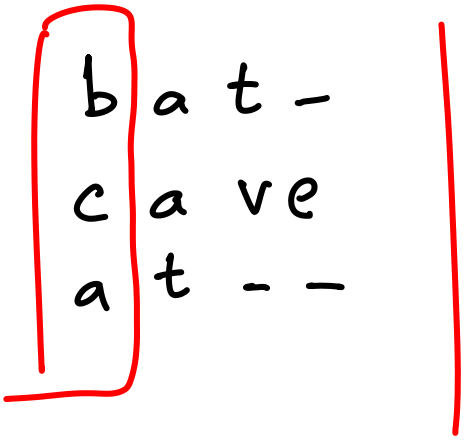
s : $|s|$: length

To apply radix, we need to align

a t _ _
b a t _
c a v e

We apply count sort to each position starting from right most





In general # rounds = L

$|L|$ = max length of string

$\Rightarrow$ for n strings we will require

$$O\left(L \cdot \left(\begin{array}{c}\text{time for}\\ \text{each round}\end{array}\right)\right)$$

$$O\left(L \cdot \left(n + |\Sigma|\right)\right)$$