# COL 702 Lecture 10 Aug 28

Dictionary ordering on strings
(Lexicographic ordering)

Eg. cave, bat, at, a, bd,
at bat cave ba,

prefix of word appears before

Can we use radix sort for lexicographic ordering?

What difference does it make if we sort from LSB → MSB vs MSB → LSB

Try to come with "inefficient" examples

Obs By appending blanks at the end of the word, we can apply normal radix sort to strings (right to left)

cave bat_ at__ _ in the smallest

First round

| ✓ | ✓ | | |
|---|---|---|---|
| bat_ | at__ | bat_ | at__ |
| at__ | bat_ | cave | bat_ |
| cave | cave | at__ | cave |

Idea: Involve only those words in round $i$ where the words have non-blank characters in $i^{th}$ position from right

Obs: If a string $s_i$ has length $l_i$ then it will be involved in $l_i$ rounds of bucket sort.

$\Rightarrow$ Cost of sorting $= \sum_i l_i$

$= \sum_j m_j$ where $m_j$ = # of strings having non-bb characters in $j^{th}$ round

$\left(\sum_j (m_j + 2^*b)\right)$

We want to implement bucket sort where # buckets can be ignored and we know in advance the non-empty buckets

If the non-empty buckets sequence is known then cost of bucket sort (count sort) is $O(n)$ where $n$ is the # of elements

c a ve $(1,c)^1$ $(2,a)^1$ $(3,v)^1$ $(4,e)^1$

b a t $(1,b)^2$ $(2,a)^2$ $(3,t)^2$

a t $(1,a)^3$ $(2,t)^3$

Let us apply sorting to the tuples

$\underbrace{(1,a)^3\ (1,b)^2\ (1,c)^1}_{m_1}\ \underbrace{(2,a)^1\ (2,a)^2\ (2,t)^3}_{m_2}\ \underbrace{(3,t)^2\ (3,v)^1}_{m_3}\ \underbrace{(4,e)}_{m_4}$

Sorting can be done in time proportional to $\sum l_i = O(N)$ $\left[O(l+N) + O(|\Sigma|+N)\right]$

( We are ignoring the cost of buckets
or we are assuming that #buckets $\leq$ #elements)

We are not paying the cost for blank char

$\Sigma$ : alphabet