Radix Sort: We can sort integers in the range \([1, n^c]\) for any constant in \(O(c \cdot n)\) steps.

Recall that in phase we were using bucket sort/count sort that takes time \(O(m + n)\)

\[
\begin{array}{c}
\text{range of} \\
\text{values}
\end{array}
\]

For the above range, we chose \(m = n\) (by making the radix \(\leq n\))

\[
\begin{array}{c}
b_1, b_2, b_3, \ldots, b_k \\
\text{range} = [1..n]
\end{array}
\]

range \((1..m_1)\) phase 1

\((1..m_2)\) phase 2
\[ \sum_i (m_i + n) \quad \text{LSB} \rightarrow \text{MSB} \]

\[ \text{range of values in phase } i \]

\[ \sum_i \mid \beta_i \mid = N \quad \mid \beta_i \mid = t_i \]

\[ \text{input size} \]

\[ \text{map } \mid \beta_i \mid = L \]

at, ate, net, class

Comparison: \( O (n \log n \cdot L) \)

\[ O \left( L \cdot (n + |B|) \right) \]

\[ O \left( L \cdot n \right) \quad |x| < n \]
How good is $O(Ln) + O((N-n)n)$

Input size: $N$

$\sum li = N$

how do we make it large

to get a sense of worst case bound?

By choosing one long thing and making other things very short, say length 1

$\implies L + n-1 = N$

$L = N - n$

at $N = \frac{N}{2}$ Running time is $O(N^2)$

$X_1, X_2, X_3, \ldots, X_n$

$\begin{array}{cccccc}
& & & & & \\
& & & & & \\
1 & 2 & D & D & - & 1 \\
3 & 0 & 0 & 0 & 0 & 5
\end{array}$
Objective: To collect all elements corresponding to a portion together. And within each, we want them to be ordered according the non-empty buckets.

**Solution:** Use radix sort in the N tuples.

- Running time: $O(N + |\Sigma|) + O(N + L)$
- $1^{st}$ phase
- $2^{nd}$ phase
- Overall $O(N + \sum L \leq 1)$

$L < N$

We have all the required information to run bucket sort for each column in time proportional to the number of non-blank symbols.

If the no. of non-blank symbols in column $i$ is $N_i$, then it will take us $O(N_i)$ steps to run phase $i$ of radix sort.
The time for phase $i$ is 

$\mathcal{O}(m_i + n_i) = \mathcal{O}(n_i)$ 

# non-empty bucket

$m_i \leq n_i$

Total time $\mathcal{O}\left(\sum_i m_i \leq n_i\right) = \mathcal{O}(N)$ for radix sort.

Incl preprocessing $\mathcal{O}(N + \sum_i \leq 1)$

Issues to be resolved (as you)

1. How do we avoid the cost of moving strings (long ones) into the bucket during the bucket sort in each phase?

2. In round $i$, how do we introduce the new string of lengths $L - i$