Tries

Jonathan Windle

University of East Anglia

J.Windle@uea.ac.uk

June 2, 2017

Overview I

1. Intro
    - Example

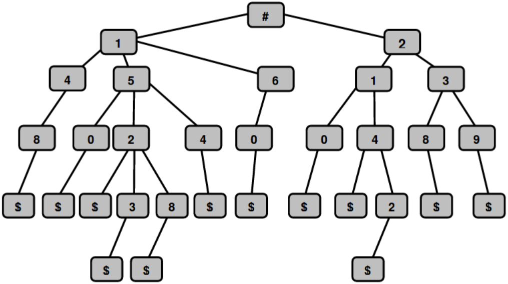
2. Linked list implementation

Intro

- A trie t for some $S \subset K$ is a tree; either it is empty or it follows these properties:
  - The root contains a special symbol: $\# \notin A$
  - Each leaf node contains a special end of key symbol: $\$ \notin A$
  - Every other node contains an element of A such that: $a_1 a_2 ... a_n \in S \quad iff \quad \# a_1 a_2 ... a_n \$$ is in path t.

Example

$A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$
$S = \{148, 150, 152, 1523, 1528, 154, 160, 210, 214, 2142, 238, 239\}$



- Every path between the root and a leaf node corresponds to a key in S.
- A trie is an appropriate representation when a combined length of all distinct prefixes in a set of keys S is small compared to the total length of all keys in S.
- Maximum number of children of a non-lead node is $m = |A| + 1$.

Linked list implementation

- First represent the trie as a binary tree:
  - Left child in the binary tree corresponds to leftmost child in the trie.
  - Right child in the binary tree corresponds to leftmost sibling in the trie.
- Keys are represented by character strings:
  - The subset of characters that can be used as symbols in a key is assumed to have been specified, as is the character to be used as the end of key symbol.
- Complexity:
  - Search is $O(nm)$ time where m is the size of the alphabet plus 1 for \$.
  - If a a branch is "Straggly" i.e. one branch has a long branch to a single key, they can be compressed and simply go to the one coalesced key.

The End