Multi-Way Search Trees
Multi-Way Search Trees

A multi-way search tree is an ordered tree such that

- Each internal node has at least two children and stores $d - 1$ key-element items $(k_i, o_i)$, where $d$ is the number of children.
- For a node with children $v_1, v_2 \ldots v_d$ storing keys $k_1, k_2 \ldots k_{d-1}$
  - keys in the subtree of $v_1$ are less than $k_1$
  - keys in the subtree of $v_i$ are between $k_{i-1}$ and $k_i$ ($i = 2, \ldots, d - 1$)
  - keys in the subtree of $v_d$ are greater than $k_{d-1}$
- The leaves store no items and serve as placeholder.
Multi-Way Inorder Traversal

We can extend the notion of inorder traversal from binary trees to multi-way search trees.

Namely, we visit item \((k_i, o_i)\) of node \(v\) between the recursive traversals of the subtrees of \(v\) rooted at children \(v_i\) and \(v_i + 1\).

An inorder traversal of a multi-way search tree visits the keys in increasing order.
Multi-Way Searching

- Similar to search in a binary search tree
- A each internal node with children $v_1 v_2 \ldots v_d$ and keys $k_1 k_2 \ldots k_{d-1}$
  - $k = k_i (i = 1, \ldots, d - 1)$: the search terminates successfully
  - $k < k_1$: we continue the search in child $v_1$
  - $k_{i-1} < k < k_i (i = 2, \ldots, d - 1)$: we continue the search in child $v_i$
  - $k > k_{d-1}$: we continue the search in child $v_d$
- Reaching an external node terminates the search unsuccessfully
- Example: search for 30