Symbol Tables

Gagnaskipan

Hjalti Magnússon (hjaltim@ru.is)

HÁSKÓLINN Í REYKJAVÍK
REYKJAVIK UNIVERSITY

23. febrúar 2015

Searching

- Important on many levels of CS
- Search by keys
  - Dictionary
  - Phone book
  - Search engine
  - Database

Symbol Table

- Aka *map*, *dictionary*, *associative array*
- A set of $(\text{key}, \text{value})$ pairs
- Operations
  - Add a new pair
  - Return an item with a given key
- Alternatively: An array (or vector) that can be indexed by anything

Implementations

- Array indexed by key
- Vector of pairs
- List of pairs
- Sorted vector of pairs
- Binary Search Tree

Skil

```
typedef Data* ValueType;
typedef int KeyType;
class Map {
    int size();
    // Adds the key to the map associated with the value.
    void add(KeyType key, ValueType value);
    // Removes the value associated with the key.
    void remove(KeyType key);
    // Retrieves the item associated with the key.
    ValueType get(KeyType key);
    // Returns true if and only if the map contains
    // the key.
    bool contains(KeyType key);
};
```

Implementations of symbol table

Method	**Get**	**Add**	**Remove**
Key-indexed array	$O(1)$	$O(1)$	$O(1)$
Unordered vector	$O(n)$	$O(1)$	$O(n)$
Unordered list	$O(n)$	$O(1)$	$O(n)$
Ordered list	$O(n)$	$O(n)$	$O(n)$
Binary search	$O(\log(n))$	$O(n)$	$O(n)$
BST (Average)	$O(\log(n))$	$O(\log(n))$	$O(\log(n))$
BST (WC)	$O(n)$	$O(n)$	$O(n)$