Lecture 17: Dictionaries

- Dictionaries
Dictionaries

- Also known as: symbol tables, maps, indices, associative arrays.

- Key-value pair abstractions that support two operations:
  - Insert a key-value pair.
  - Given a key, search for the corresponding value.

- Supported either with built-in or external libraries by the majority of programming languages.
Basic dictionary API

- **public class** Dictionary `<Key extends Comparable<Key>, Value>`
- `Dictionary()`: create an empty dictionary. By convention, values are not null.
- `void put(Key key, Value val)`: insert key-value pair.
  - Overwrites old value with new value if key already exists.
- `Value get(Key key)`: return value associated with key.
  - Returns null if key not present.
- `boolean contains(Key key)`: is there a value associated with key?
- `Iterable keys()`: all the keys in the dictionary.
- `void delete(Key key)`: delete key and associated value.
- `boolean isEmpty()`: is the dictionary empty?
- `int size()`: number of key-value pairs.
Ordered dictionaries

<table>
<thead>
<tr>
<th>keys</th>
<th>values</th>
</tr>
</thead>
<tbody>
<tr>
<td>min()</td>
<td>09:00:00 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:00:03 Phoenix</td>
</tr>
<tr>
<td></td>
<td>09:00:13 Houston</td>
</tr>
<tr>
<td>get(09:00:13)</td>
<td>09:00:59 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:01:10 Houston</td>
</tr>
<tr>
<td>floor(09:05:00)</td>
<td>09:03:13 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:10:11 Seattle</td>
</tr>
<tr>
<td>select(7)</td>
<td>09:10:25 Seattle</td>
</tr>
<tr>
<td></td>
<td>09:14:25 Phoenix</td>
</tr>
<tr>
<td></td>
<td>09:19:32 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:19:46 Chicago</td>
</tr>
<tr>
<td>keys(09:15:00, 09:25:00)</td>
<td>09:21:05 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:22:43 Seattle</td>
</tr>
<tr>
<td></td>
<td>09:22:54 Seattle</td>
</tr>
<tr>
<td></td>
<td>09:25:52 Chicago</td>
</tr>
<tr>
<td>ceiling(09:30:00)</td>
<td>09:35:21 Chicago</td>
</tr>
<tr>
<td></td>
<td>09:36:14 Seattle</td>
</tr>
<tr>
<td>max()</td>
<td>09:37:44 Phoenix</td>
</tr>
</tbody>
</table>

size(09:15:00, 09:25:00) is 5
rank(09:10:25) is 7
Ordered dictionary API

- `Key min()`: smallest key.
- `Key max()`: largest key.
- `Key floor(Key key)`: largest key less than or equal to given key.
- `Key ceiling(Key key)`: smallest key greater than or equal to given key.
- `int rank(Key key)`: number of keys less that given key.
- `Key select(int k)`: key with rank \textit{k}.
- `Iterable keys()`: all keys in dictionary in sorted order.
- `Iterable keys(int lo, int hi)`: keys in \([lo, \ldots, hi]\) in sorted order.
Printed dictionaries are all around us

- **Dictionary**: key = word, value = definition.
- **Encyclopedia**: key = term, value = article.
- **Phonebook**: key = name, value = phone number.
- **Math table**: key = math functions and input, value = function output.

**Unsupported operations:**

- Add a new key and associated value.
- Remove a given key and associated value.
- Change value associated with a given key.
Readings:

- Recommended Textbook: Chapters 3.1 (Pages 362–386)
- Website:
  - [https://algs4.cs.princeton.edu/31elementary/](https://algs4.cs.princeton.edu/31elementary/)

Practice Problems:

- 3.1.1-3.1.6