

# CS240

Nathan Sprague

November 14, 2012

# Map ADT

## Map ADT (or Dictionary or Associative Array)

- *Map()* - Creates an empty map.
- *length()* - Returns number of items in the map.
- *contains(key)* - True if the key is in the map.
- *add(key, value)* - Add a new key/value pair to the map if key is not already in the map, or modify value if key is in the map.
- *valueOf(key)* - Return the value associated with the given key.
- *remove(key)* - Remove key and its associated value from the map.
- *iterator()* - Create and return an iterator that can be used to iterate over keys.

# Python Dictionary

## Python Dictionary Methods

<i>Map()</i>	<code>dict()</code> (or <code>d = {}</code> )
<i>length()</i>	<code>__len__()</code>
<i>contains(key)</i>	<code>__contains__()</code>
<i>add(key, value)</i>	<code>__setitem__(key, value)</code>
<i>valueOf(key)</i>	<code>__getitem__(key)</code>
<i>remove(key)</i>	<code>pop(key)</code> (or <code>__del__(key)</code> )
<i>iterator()</i>	<code>__iter__()</code>

# Python List-Based Map implementation

- Fill in the following table with the worst-case big-O times.

Unsorted Python List

<i>length()</i>	
<i>add(key, value)</i> (Already in Map)	
<i>add(key, value)</i> (Not in Map)	
<i>valueOf(key)</i>	
<i>remove(key)</i>	

Sorted Python List

<i>length()</i>	
<i>add(key, value)</i> (Already in Map)	
<i>add(key, value)</i> (Not in Map)	
<i>valueOf(key)</i>	
<i>remove(key)</i>	

# Python List-Based Map implementation

- Fill in the following table with the worst-case big-O times.

Unsorted Python List

<i>length()</i>	$O(1)$
<i>add(key, value)</i> (Already in Map)	$O(n)$
<i>add(key, value)</i> (Not in Map)	$O(n)$
<i>valueOf(key)</i>	$O(n)$
<i>remove(key)</i>	$O(n)$

Sorted Python List

<i>length()</i>	$O(1)$
<i>add(key, value)</i> (Already in Map)	$O(\log n)$
<i>add(key, value)</i> (Not in Map)	$O(n)^1$
<i>valueOf(key)</i>	$O(\log n)$
<i>remove(key)</i>	$O(n)^1$

---

<sup>1</sup>Other items must be shifted.