Problem 1: LRU Cache

First. This problem asks to "*All operations must take O(1) time.*".

To solve this problem we need tu support constant time operations for **put()** and **get()**

Then I asked: what is a data structure that let's us access values in constant time? And the answer was a **Hashtable**

Then I asked: How are we going to know which of the values in the Hashmap is the least recently used? And how are we going to do operations **add()** and **remove()** quickly?
Then the answer was a **Doubly Linked List**. because if we have instant acces to a known node, we can remove it in constant time because I have the references to the **previous** and **next** nodes.

Time and Space Complexity
get --> Time O(1), Space O(1),
set --> Time O(1), Space O(1), we can add and remove items in constant time because we have the reference to the node