Hash Table

This program is for a templated hash table that stores pointers to nodes that store keys and values.

This program allows you to insert data into a table and retrieve data from the table.

To begin use #include “HashTable.h”

**Node Class Attributes**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Definition** |
| key | <K> | Holds node key |
| value | <V> | Holds node value |
| nextNode | Node<K,V>\* | Holds memory location of next Node |

**Node Class Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Return type** | **Parameter(s)** | **Definition** |
| Node() +1 overloads | Constructor | K newKey, V newValue  + Node<K,V>\* newNextNode | Used during the construction of object |
| ~Node() | Deconstructor | - | Used during deletion of object |
| getKey() | <K> |  | Return key in Node |
| setKey() | void | <K> newKey | Set key in node |
| getValue() | <V> |  | Return value in Node |
| setValue() | Void | <V> newValue | Set value in node |
| getNextNode() | Node<K,V>\* |  | Return nextNode |
| setNextNode() | void | Node<K,V>\* newNextNode | Set nextNode |

**Doubly Linked List Class Attributes**

|  |  |  |
| --- | --- | --- |
| **Name** | **Type** | **Definition** |
| tableSize | Int | Used for determining size of the array |
| table | Node<K,V>\*\* [] | Array to hold node pointers |

**Doubly Linked List Class Methods**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Return type** | **Parameter(s)** | **Definition** |
| HashTable() | Constructor | - | Used in construction of the table |
| ~HashTable() | Deconstructor | - | Used when table is deleted |
| insert() | Void | <K> keyToInsert  <V> valueToInsert | Insert a node to the table, if the key already exists change the value of that key |
| retrieve() | Node<K,V>\* | <K> keyToFind | Find a key in the table and return pointer to the object that the key is associated with. |
| makeHash() | int | <K> keyToHash | Takes in a key of any type and returns an integer associated with that key to be used in order to hash keys to the table |