class

**HashTable class**

template<typename T>;

HashTable is separate chaining hash table. The table is stored in a vector or lists and the values are index based on the returned hash value of the provided key. The elements can be accessed by providing the key and it will return the value associated with that key.

The key is set for a generic type, and all values will be returned as a string. To utilize the ToString function, classes must have their own ToString function implemented.

**Member Functions**

|  |  |
| --- | --- |
| **(constructor) default** | construct HashTable |

**Member Types**

|  |  |  |
| --- | --- | --- |
| vector<list<T>> hashLists | the lists storing keys/values | private |
| int currentSize | number of elements in the table | private - starts at 0 |

**Capacity**

|  |  |  |
| --- | --- | --- |
| Size | returns current size of the table | public |

**Modifiers**

|  |  |  |
| --- | --- | --- |
| Insert | Inserts element at index of the hash value of the key into table | public |
| Delete | deletes specified element from table | public |
| MakeEmpty | removes all items from the table | public |
| ReHash | Sets the size of the table to the next highest prime, double the current size | public |

**Element Access**

|  |  |
| --- | --- |
| Retrieve | Returns reference to the value by passing in the key |
| Contains | Returns true if the object is in the hash table |
| ToString | Returns a string of all key’s and values in the table |