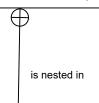


+ clearlist(): void + start() : iterator + end() : iterator + getData(int) : listType + getSize() const : int

Helper Functions

- insert(const listType, ListItem*): void
- find(int) : ListItem*
- swapItems(ListItem&, ListItem&)



iterator

- + typedef itemType : ListItem
- + typedef itemData : List::listType
- + currentPtr: const itemType*
- + itemNum : int
- + iterator()
- + iterator(itemType* start)
- + iterator(const iterator& it)
- + getNext() : itemData
- + hasNext() const : bool
- + operator*() const : itemData
- + operator=(const iterator& it): iterator&
- + operator++(): iterator&
- + operator--(): iterator&

Hashtable

- CAP: int
- nextOFbucket : int
- buckets: vector<Bucket>
- + Hashtable()
- + insertToHT(string, string)
- + getValue(string) : string
- + hashRecords(string)
- + findRecord(string): bool

Helper Functions

- hashFunc(string): int
- findRecord(string, bool, int, int)
- insertToHT(string, string, index)



0...* buckets

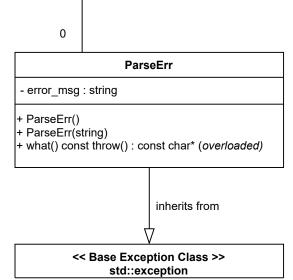
Bucket

- numSlots : const int
- nextOpenSlot : int
- count : int
- OFindex : int
- slots[numSlots] : Slots
- + Bucket()
- + Bucket(string, string)
- + Bucket(const Bucket&)
- + set(string, string, int)
- + getKey(int) : string
- + getData(int) : string
- + setKey(string, int)
- + setData(string, int)
- + hasFreeSlot(): bool + isOpen(int): bool



Slot

- key : string - data : string
- + Slot()
- + Slot(const Slot&)
- + set(string, string)
- + getKey(): string
- + getData(): string
- + setKey(string)



(from standard C++ library)

+ setData(string)