

LinkedList

1

Generated by Doxygen 1.8.18

1 Class Index	1
----------------------	----------

1 Class Index	1
1.1 Class List	1
2 File Index	1
2.1 File List	1
3 Class Documentation	2
3.1 linked_list Class Reference	2
3.1.1 Detailed Description	2
3.1.2 Constructor & Destructor Documentation	3
3.1.3 Member Function Documentation	3
3.1.4 Member Data Documentation	4
3.2 Node Class Reference	5
3.2.1 Detailed Description	5
3.2.2 Constructor & Destructor Documentation	5
3.2.3 Member Data Documentation	5
4 File Documentation	6
4.1 linked_list.cpp File Reference	6
4.2 linked_list.h File Reference	6
4.3 main.cpp File Reference	6
4.3.1 Function Documentation	6
4.4 main.h File Reference	7
4.4.1 Function Documentation	7
4.5 node.cpp File Reference	8
4.6 node.h File Reference	8
Index	9

1 Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

linked_list	2
Node	5

2 File Index

2.1 File List

Here is a list of all files with brief descriptions:

linked_list.cpp	6
linked_list.h	6
main.cpp	6
main.h	7
node.cpp	8
node.h	8

3 Class Documentation

3.1 linked_list Class Reference

```
#include <linked_list.h>
```

Public Member Functions

- **linked_list** (void)
- void **insert** (int x)
Inserts a node into the linked list.
- bool **Empty** ()
checks if linked list is empty
- void **InsertAtEnd** (int x)
adds a list item to end of list
- void **Delete** (int x)
traverses list to delete a given value.
- void **Display** ()
Displays the list values.
- int **Sum** ()
Adds all the values in the list.
- int **Average** ()
averages all values in list
- **~linked_list** ()

Private Attributes

- **Node** * **head** = NULL
- **Node** * **tail** = NULL
- **Node** * **temp** = NULL
- int **length** = 0

3.1.1 Detailed Description

Class handle the linking and searching of the LL

3.1.2 Constructor & Destructor Documentation

3.1.2.1 linked_list() `linked_list::linked_list (void)`

3.1.2.2 ~linked_list() `linked_list::~~linked_list ()`

3.1.3 Member Function Documentation

3.1.3.1 Average() `int linked_list::Average ()`

averages all values in list

Returns

average of list values

3.1.3.2 Delete() `void linked_list::Delete (int x)`

traverses list to delete a given value.

Parameters

in	x	value to delete
----	---	-----------------

3.1.3.3 Display() `void linked_list::Display ()`

Displays the list values.

3.1.3.4 Empty() `bool linked_list::Empty ()`

checks if linked list is empty

Returns

true if empty, false if it contains data

3.1.3.5 insert() `void linked_list::insert (`
`int x)`

Inserts a node into the linked list.

Parameters

in	x	value to be inserted into the list (number)
----	---	---

3.1.3.6 InsertAtEnd() `void linked_list::InsertAtEnd (`
`int x)`

adds a list item to end of list

Parameters

in	x	value to be added into the list
----	---	---------------------------------

3.1.3.7 Sum() `int linked_list::Sum ()`

Adds all the values in the list.

Returns

value of all items in the list

3.1.4 Member Data Documentation

3.1.4.1 head `Node* linked_list::head = NULL [private]`

3.1.4.2 length `int linked_list::length = 0 [private]`

3.1.4.3 tail `Node* linked_list::tail = NULL [private]`

3.1.4.4 temp `Node* linked_list::temp = NULL [private]`

The documentation for this class was generated from the following files:

- `linked_list.h`
- `linked_list.cpp`

3.2 Node Class Reference

```
#include <This>
```

Public Member Functions

- **Node** (void)
Constructs a new instance of node with value and next ptr.

Public Attributes

- `int data = 0`
- `Node * next = NULL`

3.2.1 Detailed Description

class describes a node. This class creates nodes to insert and link into the list.

3.2.2 Constructor & Destructor Documentation

3.2.2.1 Node() `Node::Node (void)`

Constructs a new instance of node with value and next ptr.

3.2.3 Member Data Documentation

3.2.3.1 data `int Node::data = 0`

3.2.3.2 next `Node* Node::next = NULL`

The documentation for this class was generated from the following files:

- `node.h`
- `node.cpp`

4 File Documentation

4.1 `linked_list.cpp` File Reference

```
#include "linked_list.h"
#include "node.h"
```

4.2 `linked_list.h` File Reference

```
#include <cstdlib>
#include <iostream>
#include <stdlib.h>
#include "node.h"
```

Classes

- class `linked_list`

4.3 `main.cpp` File Reference

```
#include "main.h"
```

Functions

- int `main ()`
- int `menu ()`
 - @brief Creates a space in memory for the program to run in*
- void `clearScreen ()`
- void `pauseScreen ()`

4.3.1 Function Documentation

4.3.1.1 clearScreen() `void clearScreen ()`

4.3.1.2 main() `int main ()`

4.3.1.3 menu() `int menu ()`

@brief Creates a space in memory for the program to run in

4.3.1.4 pauseScreen() `void pauseScreen ()`

4.4 main.h File Reference

```
#include <cstdlib>
#include <stdlib.h>
#include <iostream>
#include "node.h"
#include "linked_list.h"
```

Functions

- `int menu ()`
@brief Creates a space in memory for the program to run in
- `void clearScreen ()`
- `void pauseScreen ()`

4.4.1 Function Documentation

4.4.1.1 clearScreen() `void clearScreen ()`

4.4.1.2 menu() `int menu ()`

@brief Creates a space in memory for the program to run in

4.4.1.3 pauseScreen() `void pauseScreen ()`

4.5 node.cpp File Reference

```
#include "node.h"
```

4.6 node.h File Reference

```
#include <cstdlib>
```

Classes

- class **Node**

Index

- ~linked_list
 - linked_list, 3
- Average
 - linked_list, 3
- clearScreen
 - main.cpp, 6
 - main.h, 7
- data
 - Node, 5
- Delete
 - linked_list, 3
- Display
 - linked_list, 3
- Empty
 - linked_list, 3
- head
 - linked_list, 4
- insert
 - linked_list, 3
- InsertAtEnd
 - linked_list, 4
- length
 - linked_list, 4
- linked_list, 2
 - ~linked_list, 3
 - Average, 3
 - Delete, 3
 - Display, 3
 - Empty, 3
 - head, 4
 - insert, 3
 - InsertAtEnd, 4
 - length, 4
 - linked_list, 3
 - Sum, 4
 - tail, 4
 - temp, 4
- linked_list.cpp, 6
- linked_list.h, 6
- main
 - main.cpp, 7
- main.cpp, 6
 - clearScreen, 6
 - main, 7
 - menu, 7
 - pauseScreen, 7
- main.h, 7
 - clearScreen, 7
 - menu, 7
 - pauseScreen, 7
- menu
 - main.cpp, 7
 - main.h, 7
- next
 - Node, 5
- Node, 5
 - data, 5
 - next, 5
 - Node, 5
- node.cpp, 8
- node.h, 8
- pauseScreen
 - main.cpp, 7
 - main.h, 7
- Sum
 - linked_list, 4
- tail
 - linked_list, 4
- temp
 - linked_list, 4