

gagchat

0.1

Generated by Doxygen 1.8.17

| | |
|--|-----------|
| 1 gagchat | 1 |
| 2 Class Index | 3 |
| 2.1 Class List | 3 |
| 3 File Index | 5 |
| 3.1 File List | 5 |
| 4 Class Documentation | 7 |
| 4.1 RBTree Struct Reference | 7 |
| 4.2 SBSTnode Struct Reference | 8 |
| 4.3 SBucket Struct Reference | 8 |
| 4.3.1 Detailed Description | 8 |
| 4.4 SHashTable Struct Reference | 9 |
| 4.5 SHLNode Struct Reference | 9 |
| 4.6 SMessage Struct Reference | 10 |
| 4.7 STCPSocket Struct Reference | 10 |
| 4.8 SThreadArgs Struct Reference | 11 |
| 4.8.1 Detailed Description | 11 |
| 4.9 TInfoHashtable Struct Reference | 11 |
| 4.9.1 Detailed Description | 12 |
| 4.10 TQueue Struct Reference | 12 |
| 4.10.1 Detailed Description | 12 |
| 4.11 Treenode Struct Reference | 13 |
| 5 File Documentation | 15 |
| 5.1 include/client.h File Reference | 15 |
| 5.1.1 Detailed Description | 16 |
| 5.1.2 Function Documentation | 17 |
| 5.1.2.1 client_message_receiver() | 17 |
| 5.2 include/server.h File Reference | 17 |
| 5.2.1 Detailed Description | 19 |
| 5.2.2 Typedef Documentation | 19 |
| 5.2.2.1 ThreadArgs | 19 |
| 5.2.3 Function Documentation | 20 |
| 5.2.3.1 server_manage_client() | 20 |
| 5.2.3.2 server_sendToAll() | 20 |
| 5.2.3.3 server_sendUsersTo() | 20 |
| 5.3 library/include/gui.h File Reference | 20 |
| 5.3.1 Detailed Description | 23 |
| 5.3.2 Function Documentation | 23 |
| 5.3.2.1 gui_print_message() | 23 |
| 5.3.2.2 gui_set_color() | 23 |
| 5.4 library/include/hashtable.h File Reference | 24 |

| | |
|--|----|
| 5.4.1 Detailed Description | 26 |
| 5.4.2 Typedef Documentation | 26 |
| 5.4.2.1 TBucket | 26 |
| 5.4.3 Function Documentation | 26 |
| 5.4.3.1 hashTableCreate() | 26 |
| 5.4.3.2 hashTableDelete() | 26 |
| 5.4.3.3 hashTableDestroy() | 27 |
| 5.4.3.4 hashTableInsert() | 27 |
| 5.4.3.5 hashTablePrint() | 27 |
| 5.4.3.6 hashTableResize() | 28 |
| 5.4.3.7 hashTableSearch() | 28 |
| 5.4.3.8 keyHash() | 29 |
| 5.4.3.9 keyHashD() | 29 |
| 5.4.3.10 keyHashExpand() | 29 |
| 5.5 library/include/infoHashtable.h File Reference | 30 |
| 5.5.1 Detailed Description | 31 |
| 5.5.2 Typedef Documentation | 31 |
| 5.5.2.1 TKeyHashtable | 31 |
| 5.5.3 Function Documentation | 32 |
| 5.5.3.1 infoEqual() | 32 |
| 5.5.3.2 infoGreater() | 32 |
| 5.5.3.3 infoLess() | 32 |
| 5.5.3.4 infoPrint() | 33 |
| 5.6 library/include/infoList.h File Reference | 33 |
| 5.6.1 Detailed Description | 34 |
| 5.6.2 Typedef Documentation | 34 |
| 5.6.2.1 TInfoList | 34 |
| 5.6.3 Function Documentation | 34 |
| 5.6.3.1 infoListEqual() | 35 |
| 5.6.3.2 infoListPrint() | 35 |
| 5.7 library/include/list.h File Reference | 35 |
| 5.7.1 Detailed Description | 37 |
| 5.7.2 Typedef Documentation | 37 |
| 5.7.2.1 THLNode | 37 |
| 5.7.3 Function Documentation | 37 |
| 5.7.3.1 listCreate() | 37 |
| 5.7.3.2 listDelete() | 37 |
| 5.7.3.3 listDestroy() | 38 |
| 5.7.3.4 listInsert() | 38 |
| 5.7.3.5 listPrint() | 39 |
| 5.7.3.6 listSearch() | 39 |
| 5.8 library/include/message.h File Reference | 39 |

| | |
|--|----|
| 5.8.1 Detailed Description | 40 |
| 5.8.2 Macro Definition Documentation | 41 |
| 5.8.2.1 MESSAGE | 41 |
| 5.8.2.2 MESSAGE_STRUCT | 41 |
| 5.8.3 Function Documentation | 41 |
| 5.8.3.1 message_code_constructor() | 41 |
| 5.8.3.2 message_constructor() | 41 |
| 5.8.3.3 message_create() | 42 |
| 5.8.3.4 message_destroy() | 42 |
| 5.9 library/include/queue.h File Reference | 42 |
| 5.9.1 Detailed Description | 44 |
| 5.10 library/include/redblackTree.h File Reference | 44 |
| 5.10.1 Detailed Description | 45 |
| 5.11 library/include/TBST.h File Reference | 46 |
| 5.11.1 Detailed Description | 47 |
| 5.12 library/include/tcp_socket.h File Reference | 48 |
| 5.12.1 Detailed Description | 49 |
| 5.12.2 Function Documentation | 50 |
| 5.12.2.1 tcp_socket_client_connect() | 50 |
| 5.12.2.2 tcp_socket_create() | 50 |
| 5.12.2.3 tcp_socket_destroy() | 50 |
| 5.12.2.4 tcp_socket_recv_message() | 51 |
| 5.12.2.5 tcp_socket_send_message() | 51 |
| 5.12.2.6 tcp_socket_server_accept() | 51 |
| 5.12.2.7 tcp_socket_server_listen() | 52 |
| 5.13 library/include/TInfoBST.h File Reference | 52 |
| 5.13.1 Detailed Description | 53 |
| 5.14 library/include/treeNode.h File Reference | 53 |
| 5.14.1 Detailed Description | 55 |
| 5.15 library/src/gui.c File Reference | 55 |
| 5.15.1 Detailed Description | 56 |
| 5.15.2 Function Documentation | 56 |
| 5.15.2.1 gui_print_horizontal_line() | 57 |
| 5.15.2.2 gui_print_horizontal_msg_line() | 57 |
| 5.15.2.3 gui_print_message() | 57 |
| 5.15.2.4 gui_print_n_times() | 57 |
| 5.15.2.5 gui_print_user_msg() | 58 |
| 5.15.2.6 gui_set_color() | 58 |
| 5.16 library/src/hashtable.c File Reference | 58 |
| 5.16.1 Detailed Description | 59 |
| 5.16.2 Function Documentation | 60 |
| 5.16.2.1 hashTableCreate() | 60 |

| | | |
|-----------|---|----|
| 5.16.2.2 | hashTableDelete() | 60 |
| 5.16.2.3 | hashTableDestroy() | 61 |
| 5.16.2.4 | hashTableInsert() | 61 |
| 5.16.2.5 | hashTablePrint() | 61 |
| 5.16.2.6 | hashTableResize() | 61 |
| 5.16.2.7 | hashTableSearch() | 62 |
| 5.16.2.8 | keyHash() | 62 |
| 5.16.2.9 | keyHashD() | 63 |
| 5.16.2.10 | keyHashExpand() | 63 |
| 5.17 | library/src/infoList.c File Reference | 63 |
| 5.17.1 | Detailed Description | 64 |
| 5.17.2 | Function Documentation | 64 |
| 5.17.2.1 | infoListEqual() | 64 |
| 5.17.2.2 | infoListPrint() | 65 |
| 5.18 | library/src/list.c File Reference | 65 |
| 5.18.1 | Detailed Description | 66 |
| 5.18.2 | Function Documentation | 66 |
| 5.18.2.1 | listCreate() | 66 |
| 5.18.2.2 | listDelete() | 66 |
| 5.18.2.3 | listDestroy() | 67 |
| 5.18.2.4 | listInsert() | 67 |
| 5.18.2.5 | listPrint() | 68 |
| 5.18.2.6 | listSearch() | 68 |
| 5.18.2.7 | nodeCreate() | 68 |
| 5.18.2.8 | nodeDestroy() | 69 |
| 5.19 | library/src/message.c File Reference | 69 |
| 5.19.1 | Detailed Description | 69 |
| 5.19.2 | Function Documentation | 70 |
| 5.19.2.1 | message_code_constructor() | 70 |
| 5.19.2.2 | message_constructor() | 71 |
| 5.19.2.3 | message_create() | 71 |
| 5.19.2.4 | message_destroy() | 71 |
| 5.20 | library/src/queue.c File Reference | 71 |
| 5.20.1 | Detailed Description | 72 |
| 5.21 | library/src/redblackTree.c File Reference | 73 |
| 5.21.1 | Detailed Description | 73 |
| 5.22 | library/src/TBST.c File Reference | 74 |
| 5.22.1 | Detailed Description | 75 |
| 5.23 | library/src/tcp_socket.c File Reference | 75 |
| 5.23.1 | Detailed Description | 76 |
| 5.23.2 | Function Documentation | 76 |
| 5.23.2.1 | tcp_socket_client_connect() | 76 |

| | |
|--|-----------|
| 5.23.2.2 tcp_socket_create() | 77 |
| 5.23.2.3 tcp_socket_destroy() | 77 |
| 5.23.2.4 tcp_socket_recv_message() | 77 |
| 5.23.2.5 tcp_socket_send_message() | 78 |
| 5.23.2.6 tcp_socket_server_accept() | 78 |
| 5.23.2.7 tcp_socket_server_listen() | 79 |
| 5.24 library/src/TInfoBST.c File Reference | 79 |
| 5.24.1 Detailed Description | 79 |
| 5.25 library/src/treeNode.c File Reference | 80 |
| 5.25.1 Detailed Description | 81 |
| 5.26 library/test.c File Reference | 82 |
| 5.26.1 Detailed Description | 82 |
| 5.27 src/client.c File Reference | 83 |
| 5.27.1 Detailed Description | 83 |
| 5.27.2 Function Documentation | 84 |
| 5.27.2.1 client_message_receiver() | 84 |
| 5.28 src/server.c File Reference | 84 |
| 5.28.1 Detailed Description | 85 |
| 5.28.2 Function Documentation | 85 |
| 5.28.2.1 server_manage_client() | 86 |
| 5.28.2.2 server_sendToAll() | 86 |
| 5.28.2.3 server_sendUsersTo() | 86 |
| Index | 87 |

Chapter 1

gagchat

A simple implementation of a common chat program in C.

Chapter 2

Class Index

2.1 Class List

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

| | |
|--------------------------------|----|
| RBTree | 7 |
| SBSTnode | 8 |
| SBucket | |
| Struct of Hashtable | 8 |
| SHashTable | 9 |
| SHLNode | 9 |
| SMessage | 10 |
| STCPSocket | 10 |
| SThreadArgs | |
| Struct exchanged by Threads | 11 |
| TInfoHashtable | |
| Definition of HashTable | 11 |
| TQueue | |
| Struct of the queue | 12 |
| Treenode | 13 |

Chapter 3

File Index

3.1 File List

Here is a list of all documented files with brief descriptions:

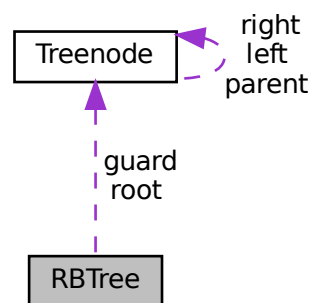
| | |
|---|----|
| include/ client.h | |
| Client header file | 15 |
| include/ server.h | |
| Server header file | 17 |
| library/ test.c | 82 |
| library/include/ gui.h | 20 |
| library/include/ hashtable.h | 24 |
| library/include/ infoHashtable.h | 30 |
| library/include/ infoList.h | 33 |
| library/include/ list.h | 35 |
| library/include/ message.h | |
| Definition of struct Message | 39 |
| library/include/ queue.h | 42 |
| library/include/ redblackTree.h | 44 |
| library/include/ TBST.h | 46 |
| library/include/ tcp_socket.h | |
| Implementation of tcp socket both for client and server | 48 |
| library/include/ TInfoBST.h | 52 |
| library/include/ treeNode.h | 53 |
| library/src/ gui.c | 55 |
| library/src/ hashtable.c | 58 |
| library/src/ infoList.c | 63 |
| library/src/ list.c | 65 |
| library/src/ message.c | 69 |
| library/src/ queue.c | 71 |
| library/src/ redblackTree.c | 73 |
| library/src/ TBST.c | 74 |
| library/src/ tcp_socket.c | 75 |
| library/src/ TInfoBST.c | 79 |
| library/src/ treeNode.c | 80 |
| src/ client.c | |
| Client Implementation of gagchat | 83 |
| src/ server.c | |
| Server implementation of gagchat | 84 |

Chapter 4

Class Documentation

4.1 RBTREE Struct Reference

Collaboration diagram for RBTREE:



Public Attributes

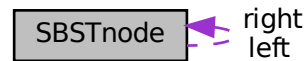
- [Treenode](#) * **root**
- [Treenode](#) * **guard**
- int(* **comparator**)(const void *, const void *)

The documentation for this struct was generated from the following file:

- library/include/[redblackTree.h](#)

4.2 SBSTnode Struct Reference

Collaboration diagram for SBSTnode:



Public Attributes

- TInfoBST **info**
- struct [SBSTnode](#) * **left**
- struct [SBSTnode](#) * **right**

The documentation for this struct was generated from the following file:

- library/include/[TBST.h](#)

4.3 SBucket Struct Reference

Struct of Hashtable.

```
#include <hashtable.h>
```

Public Attributes

- [TKeyHashtable](#) **key**
- TValueHashtable **value**

4.3.1 Detailed Description

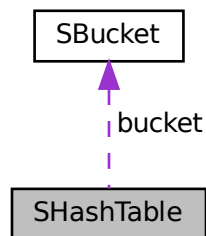
Struct of Hashtable.

The documentation for this struct was generated from the following file:

- library/include/[hashtable.h](#)

4.4 SHashTable Struct Reference

Collaboration diagram for SHashTable:



Public Attributes

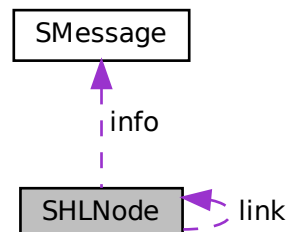
- [TBucket](#) * **bucket**
- int * **used**
- int **n_bucket**
- int **n_used**

The documentation for this struct was generated from the following file:

- library/include/[hashtable.h](#)

4.5 SHLNode Struct Reference

Collaboration diagram for SHLNode:



Public Attributes

- [TInfoList](#) **info**
- [THLNode](#) * **link**

The documentation for this struct was generated from the following file:

- [library/include/list.h](#)

4.6 SMessage Struct Reference

Public Attributes

- char **user** [USR_MAXLEN]
- char **text** [TXT_MAXLEN]
- int **code**

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

- [library/include/infoList.h](#)
- [library/include/message.h](#)

4.7 STCPSocket Struct Reference

Public Attributes

- int **sockfd**
- struct sockaddr_in **servaddr**

The documentation for this struct was generated from the following file:

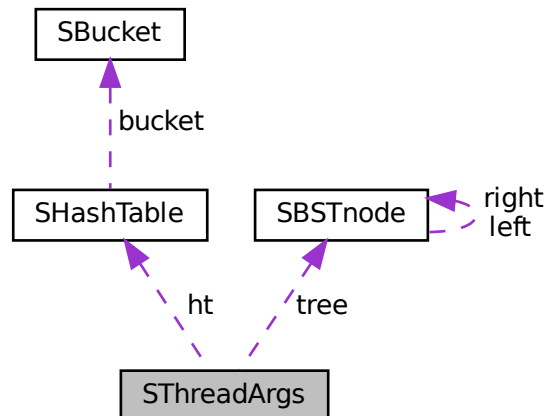
- [library/include/tcp_socket.h](#)

4.8 SThreadArgs Struct Reference

Struct exchanged by Threads.

```
#include <server.h>
```

Collaboration diagram for SThreadArgs:



Public Attributes

- [THashTable](#) * **ht**
- char * **selectedUser**
- pthread_mutex_t * **lock**
- int **connection_fd**
- [TBST](#) * **tree**

4.8.1 Detailed Description

Struct exchanged by Threads.

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

- include/[client.h](#)
- include/[server.h](#)

4.9 TInfoHashtable Struct Reference

definition of HashTable

```
#include <infoHashtable.h>
```

Public Attributes

- [TKeyHashtable](#) **key**
- TValueHashtable **value**

4.9.1 Detailed Description

definition of HashTable

The documentation for this struct was generated from the following file:

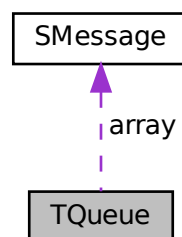
- library/include/[infoHashtable.h](#)

4.10 TQueue Struct Reference

Struct of the queue.

```
#include <queue.h>
```

Collaboration diagram for TQueue:



Public Attributes

- int **n_element**
- int **front**
- int **back**
- int **size**
- [Message](#) ** **array**

4.10.1 Detailed Description

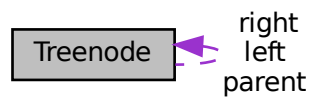
Struct of the queue.

The documentation for this struct was generated from the following file:

- library/include/[queue.h](#)

4.11 Treenode Struct Reference

Collaboration diagram for Treenode:



Public Attributes

- void * **data**
- char **color**
- struct [Treenode](#) * **left**
- struct [Treenode](#) * **right**
- struct [Treenode](#) * **parent**

The documentation for this struct was generated from the following file:

- library/include/[treeNode.h](#)

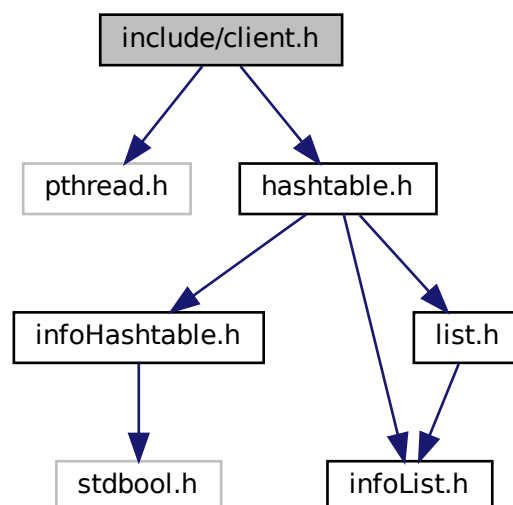
Chapter 5

File Documentation

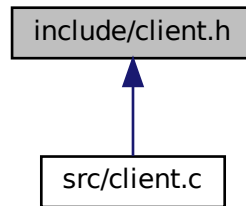
5.1 include/client.h File Reference

Client header file.

```
#include <pthread.h>
#include "hashtable.h"
Include dependency graph for client.h:
```



This graph shows which files directly or indirectly include this file:



Classes

- struct [SThreadArgs](#)
Struct exchanged by Threads.

Macros

- `#define QUEUE_SIZE 32`

Typedefs

- typedef struct [SThreadArgs](#) **ThreadArgs**

Functions

- void * [client_message_receiver](#) (void *arg)
thread for message receiving.

5.1.1 Detailed Description

Client header file.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-28

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.1.2 Function Documentation

5.1.2.1 client_message_receiver()

```
void* client_message_receiver (
    void * arg )
```

thread for message receiving.

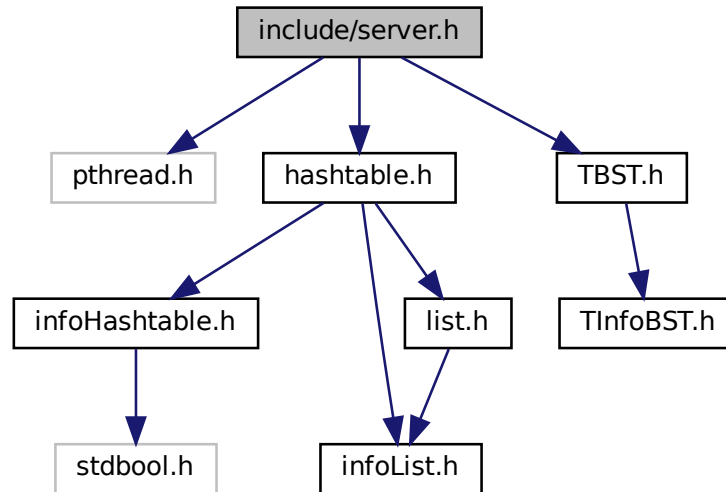
5.2 include/server.h File Reference

Server header file.

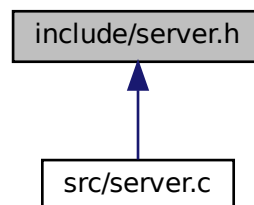
```
#include <pthread.h>
#include "hashtable.h"
```

```
#include "TBST.h"
```

Include dependency graph for server.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [SThreadArgs](#)
Struct exchanged by Threads.

Typedefs

- typedef struct [SThreadArgs](#) ThreadArgs
Struct exchanged by Threads.

Functions

- void `server_sendToAll` (`THashTable` *ht, `TBST` tree, `Message` *message)
Sends a message to all clients.
- void `server_sendUsersTo` (int connection_fd, `TBST` users)
Sends the users list to specified client.
- void * `server_manage_client` (void *arg)
Thread for managing a generic client.

5.2.1 Detailed Description

Server header file.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-31

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.2.2 Typedef Documentation

5.2.2.1 ThreadArgs

```
typedef struct SThreadArgs ThreadArgs
```

Struct exchanged by Threads.

5.2.3 Function Documentation

5.2.3.1 server_manage_client()

```
void* server_manage_client (
    void * arg )
```

Thread for managing a generic client.

5.2.3.2 server_sendToAll()

```
void server_sendToAll (
    THashTable * ht,
    TBST tree,
    Message * message )
```

Sends a message to all clients.

Parameters

| | |
|----------------|---------------------------------------|
| <i>ht</i> | the hashtable containing connections. |
| <i>tree</i> | tree list of users. |
| <i>message</i> | the message to send. |

5.2.3.3 server_sendUsersTo()

```
void server_sendUsersTo (
    int connection_fd,
    TBST users )
```

Sends the users list to specified client.

Parameters

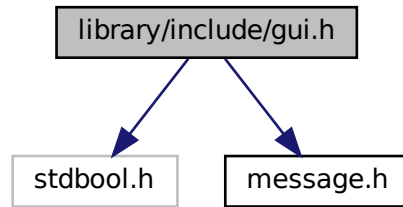
| | |
|-----------------------------------|-----------------------------|
| <i>connection</i> ↔ <i>_fd</i> | connection file descriptor. |
| <i>users</i> | tree with all of users. |

5.3 library/include/gui.h File Reference

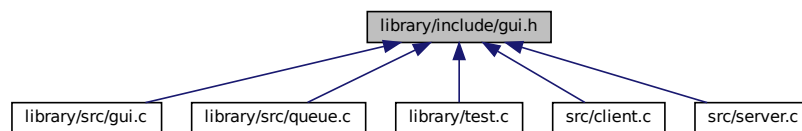
```
#include <stdbool.h>
```

```
#include "message.h"
```

Include dependency graph for gui.h:



This graph shows which files directly or indirectly include this file:



Macros

- `#define WIDHT_SPACE 15`
- `#define COLUMNS 70`
- `#define ROWS 20`
- `#define Color_Off "\033[0m"`
- `#define Default_Color "\033[0m"`
- `#define Black "\033[30m"`
- `#define Red "\033[31m"`
- `#define Green "\033[32m"`
- `#define Yellow "\033[33m"`
- `#define Blue "\033[34m"`
- `#define Purple "\033[35m"`
- `#define Cyan "\033[36m"`
- `#define White "\033[37m"`
- `#define BBlack "\033[1;30m"`
- `#define BRed "\033[1;31m"`
- `#define BGreen "\033[1;32m"`
- `#define BYellow "\033[1;33m"`
- `#define BBlue "\033[1;34m"`
- `#define BPurple "\033[1;35m"`
- `#define BCyan "\033[1;36m"`
- `#define BWhite "\033[1;37m"`
- `#define UBlack "\033[4;30m"`

- `#define URed "\033[4;31m"`
- `#define UGreen "\033[4;32m"`
- `#define UYellow "\033[4;33m"`
- `#define UBlue "\033[4;34m"`
- `#define UPurple "\033[4;35m"`
- `#define UCyan "\033[4;36m"`
- `#define UWhite "\033[4;37m"`
- `#define On_Black "\033[40m"`
- `#define On_Red "\033[41m"`
- `#define On_Green "\033[42m"`
- `#define On_Yellow "\033[43m"`
- `#define On_Blue "\033[44m"`
- `#define On_Purple "\033[45m"`
- `#define On_Cyan "\033[46m"`
- `#define On_White "\033[47m"`
- `#define IBlack "\033[90m"`
- `#define IRed "\033[91m"`
- `#define IGreen "\033[92m"`
- `#define IYellow "\033[93m"`
- `#define IBlue "\033[94m"`
- `#define IPurple "\033[95m"`
- `#define ICyan "\033[96m"`
- `#define IWhite "\033[97m"`
- `#define BIBlack "\033[1;90m"`
- `#define BIRed "\033[1;91m"`
- `#define BIGreen "\033[1;92m"`
- `#define BIYellow "\033[1;93m"`
- `#define BIBBlue "\033[1;94m"`
- `#define BIPurple "\033[1;95m"`
- `#define BICyan "\033[1;96m"`
- `#define BIWhite "\033[1;97m"`
- `#define On_IBlack "\033[100m"`
- `#define On_IRed "\033[101m"`
- `#define On_IGreen "\033[102m"`
- `#define On_IYellow "\033[103m"`
- `#define On_IBBlue "\033[104m"`
- `#define On_IPurple "\033[105m"`
- `#define On_ICyan "\033[106m"`
- `#define On_IWhite "\033[107m"`

Functions

- void `gui_set_color` (char *color)
set the color of text or background, apply a text style
- void `gui_print_message` (Message *msg, bool received)
- void `gui_print_menu` (char *user)
- void `gui_print_list_users_header` ()
- void `gui_print_list_user` (char *user)
- void `gui_print_list_users_footer` ()
- void `gui_clear_Screen` ()

5.3.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-11-03

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.3.2 Function Documentation

5.3.2.1 `gui_print_message()`

```
void gui_print_message (
    Message * msg,
    bool received )
```

Parameters

| | |
|-----------------|--|
| <i>msg</i> | |
| <i>received</i> | |

5.3.2.2 `gui_set_color()`

```
void gui_set_color (
```

```
char * color )
```

set the color of text or background, apply a text style

Parameters

| | |
|--------------|--|
| <i>color</i> | |
|--------------|--|

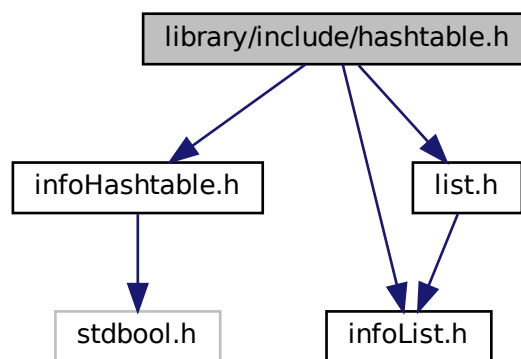
5.4 library/include/hashtable.h File Reference

```
#include "infoHashtable.h"
```

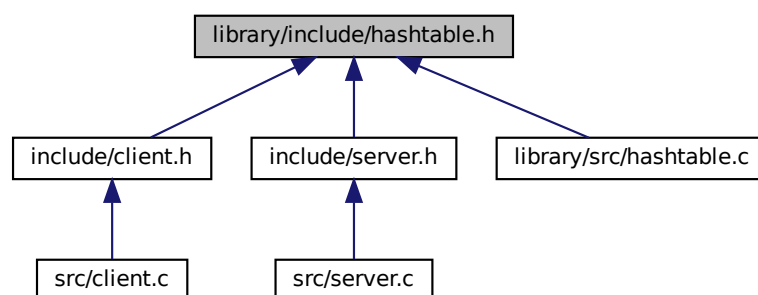
```
#include "infoList.h"
```

```
#include "list.h"
```

Include dependency graph for hashtable.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [SBucket](#)
Struct of Hashtable.
- struct [SHashTable](#)

Macros

- #define **FNV_OFFSET** 14695981039346656037UL
- #define **FNV_PRIME** 1099511628211UL
- #define **FNV_PRIME_2** 10995116282UL
- #define **CAPACITY** 64
- #define **MAX_LOAD** 0.75
- #define **GROW_FACTOR** 2
- #define **MAX_KEY_SIZE** 64

Typedefs

- typedef struct [SBucket](#) **TBucket**
Struct of Hashtable.
- typedef struct [SHashTable](#) **THashTable**

Functions

- [THashTable *](#) [hashTableCreate](#) ()
Creation of a new HashTable.
- void [hashTableResize](#) ([THashTable](#) *ht, int n)
Resize hashtable.
- void [hashTableDestroy](#) ([THashTable](#) *ht)
destroy the HashTable
- [TValueHashtable](#) [hashTableSearch](#) ([THashTable](#) *ht, [TKeyHashtable](#) key)
search in a HashTable
- void [hashTableInsert](#) ([THashTable](#) *ht, [TKeyHashtable](#) key, [TValueHashtable](#) value)
Insert a new key and value in a HashTable.
- void [hashTableDelete](#) ([THashTable](#) *ht, [TKeyHashtable](#) key)
Delete a key in a HashTable.
- void [hashTablePrint](#) ([THashTable](#) *ht)
Print a HashTable.
- [u_int64_t](#) [keyHash](#) ([TKeyHashtable](#))
hash a key (function h(x))
- [u_int64_t](#) [keyHashD](#) ([TKeyHashtable](#))
hash a key (function d(x))
- [u_int64_t](#) [keyHashExpand](#) ([TKeyHashtable](#), [u_int64_t](#), int)
*double hashing return hash = h(x) + j*d(x)*

5.4.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.4.2 Typedef Documentation

5.4.2.1 TBucket

```
typedef struct SBucket TBucket
```

Struct of Hashtable.

5.4.3 Function Documentation

5.4.3.1 hashTableCreate()

```
THashTable* hashTableCreate ( )
```

Creation of a new HashTable.

Returns

THashTable*

5.4.3.2 hashTableDelete()

```
void hashTableDelete (
    THashTable * ht,
    TKeyHashtable key )
```

Delete a key in a HashTable.

Parameters

| | |
|------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to delete |

5.4.3.3 hashtableDestroy()

```
void hashtableDestroy (
    THashTable * ht )
```

destroy the HashTable

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
|-----------|----------------------|

5.4.3.4 hashtableInsert()

```
void hashtableInsert (
    THashTable * ht,
    TKeyHashtable key,
    TValueHashtable value )
```

Insert a new key and value in a HashTable.

Parameters

| | |
|--------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to insert |
| <i>value</i> | value to insert |

5.4.3.5 hashtablePrint()

```
void hashtablePrint (
    THashTable * ht )
```

Print a HashTable.

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
|-----------|----------------------|

5.4.3.6 hashTableResize()

```
void hashTableResize (
    THashTable * ht,
    int n )
```

Resize hashtable.

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>n</i> | new size |

Resize hashtable.

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>n</i> | new size |

5.4.3.7 hashTableSearch()

```
TValueHashtable hashTableSearch (
    THashTable * ht,
    TKeyHashtable key )
```

search in a HashTable

Parameters

| | |
|------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to find |

Returns

TValueHashtable

search in a HashTable

Parameters

| | |
|------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to find |

Returns

TValueHashtable

5.4.3.8 keyHash()

```
u_int64_t keyHash (
    TKeyHashtable key )
```

hash a key (function h(x))

Returns

u_int64_t

5.4.3.9 keyHashD()

```
u_int64_t keyHashD (
    TKeyHashtable key )
```

hash a key (function d(x))

Returns

u_int64_t

5.4.3.10 keyHashExpande()

```
u_int64_t keyHashExpande (
    TKeyHashtable key,
    u_int64_t hash,
    int j )
```

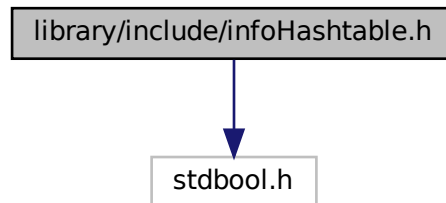
double hashing return $\text{hash} = h(x) + j*d(x)$ **Returns**

u_int64_t

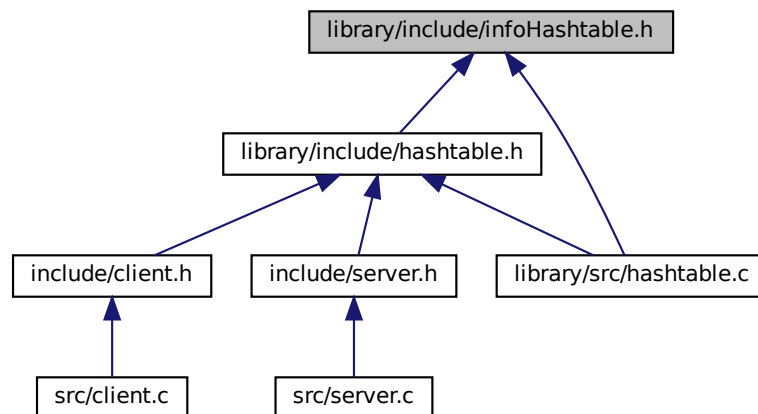
5.5 library/include/infoHashtable.h File Reference

```
#include "stdbool.h"
```

Include dependency graph for infoHashtable.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [TInfoHashtable](#)
definition of HashTable

Typedefs

- typedef char * [TKeyHashtable](#)
type of HashTable
- typedef void * [TValueHashtable](#)

Functions

- bool `infoEqual` (`TInfoHashtable`, `TInfoHashtable`)
check if first value is equal to the second value
- bool `infoGreater` (`TInfoHashtable`, `TInfoHashtable`)
check if first value is greater to the second value
- bool `infoLess` (`TInfoHashtable`, `TInfoHashtable`)
check if first value is less to the second value
- void `infoPrint` (`TInfoHashtable`)
structured print of the value

5.5.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.5.2 Typedef Documentation

5.5.2.1 TKeyHashtable

```
typedef char* TKeyHashtable
```

type of HashTable

5.5.3 Function Documentation

5.5.3.1 infoEqual()

```
bool infoEqual (
    TInfoHashtable val1,
    TInfoHashtable val2 )
```

check if frist value is equal to the second value

Returns

bool

5.5.3.2 infoGreater()

```
bool infoGreater (
    TInfoHashtable val1,
    TInfoHashtable val2 )
```

check if frist value is greater to the second value

Returns

bool

5.5.3.3 infoLess()

```
bool infoLess (
    TInfoHashtable val1,
    TInfoHashtable val2 )
```

check if frist value is less to the second value

Returns

bool

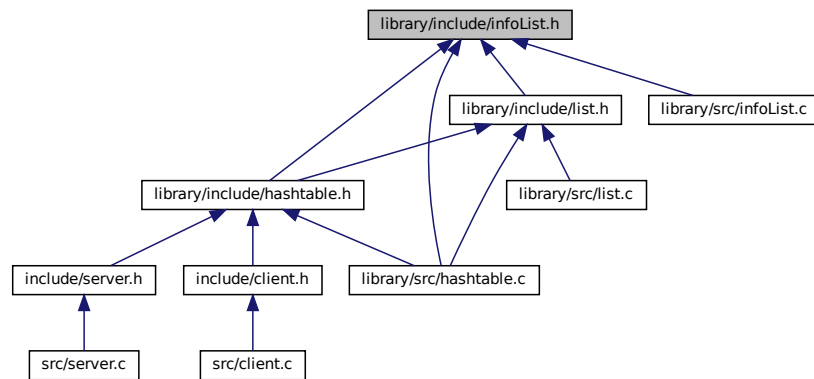
5.5.3.4 infoPrint()

```
void infoPrint (
    TInfoHashtable val1 )
```

structured print of the value

5.6 library/include/infoList.h File Reference

This graph shows which files directly or indirectly include this file:



Classes

- struct [SMessage](#)

Macros

- #define **USR_MAXLEN** 64
- #define **TXT_MAXLEN** 1024
- #define **MESSAGE_STRUCT**

Typedefs

- typedef struct [SMessage](#) **Message**
- typedef [Message](#) **TInfoList**
type of List

Functions

- int [infoListEqual](#) ([TInfoList](#), [TInfoList](#))
check if all the information of frist value are the same of the second value
- void [infoListPrint](#) ([TInfoList](#))
structured print of the value

5.6.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.6.2 Typedef Documentation

5.6.2.1 TInfoList

```
typedef Message TInfoList
```

type of List

5.6.3 Function Documentation

5.6.3.1 infoListEqual()

```
int infoListEqual (
    TInfoList info1,
    TInfoList info2 )
```

check if all the information of frist value are the same of the second value

Returns

bool

5.6.3.2 infoListPrint()

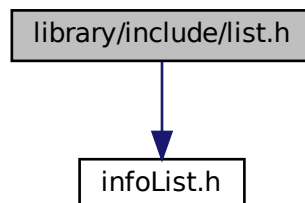
```
void infoListPrint (
    TInfoList info )
```

structured print of the value

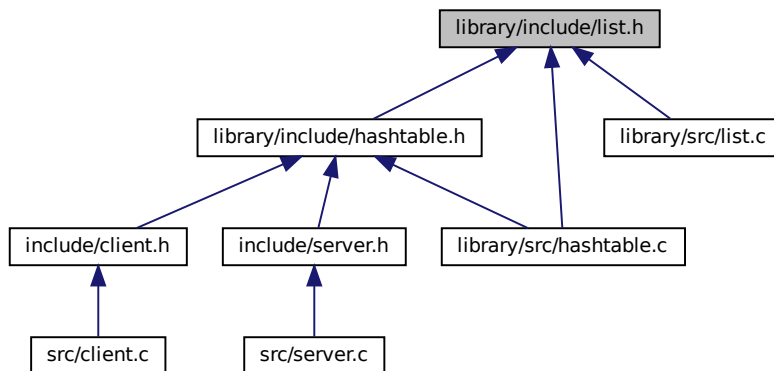
5.7 library/include/list.h File Reference

```
#include "infoList.h"
```

Include dependency graph for list.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [SHLNode](#)

Typedefs

- typedef struct [SHLNode](#) [THLNode](#)
Struct of List.
- typedef [THLNode](#) * [TList](#)

Functions

- [TList](#) [listCreate](#) ()
Creation of a new List.
- [TList](#) [listDestroy](#) ([TList](#) list)
Destroy a List.
- [TList](#) [listInsert](#) ([TList](#) list, [TInfoList](#) info)
Insert a new value in the List.
- [THLNode](#) * [listSearch](#) ([TList](#) list, [TInfoList](#) info)
Search an element into the List.
- [TList](#) [listDelete](#) ([TList](#) list, [TInfoList](#) info)
Delete an element from a list.
- void [listPrint](#) ([TList](#) list)
Print of an entire List.

5.7.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.7.2 Typedef Documentation

5.7.2.1 THLNode

```
typedef struct SHLNode THLNode
```

Struct of List.

5.7.3 Function Documentation

5.7.3.1 listCreate()

```
TList listCreate ( )
```

Creation of a new List.

Returns

TList

5.7.3.2 listDelete()

```
TList listDelete (
    TList list,
    TInfoList info )
```

Delete an element from a list.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

Returns

TList

5.7.3.3 listDestroy()

```
TList listDestroy (  
    TList list )
```

Destroy a List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
|-------------|-----------------|

Returns

TList

5.7.3.4 listInsert()

```
TList listInsert (  
    TList l,  
    TInfoList info )
```

Insert a new value in the List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

Returns

TList

5.7.3.5 listPrint()

```
void listPrint (
    TList list )
```

Print of an entire List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
|-------------|-----------------|

5.7.3.6 listSearch()

```
THLNode* listSearch (
    TList list,
    TInfoList info )
```

Search an element into the List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

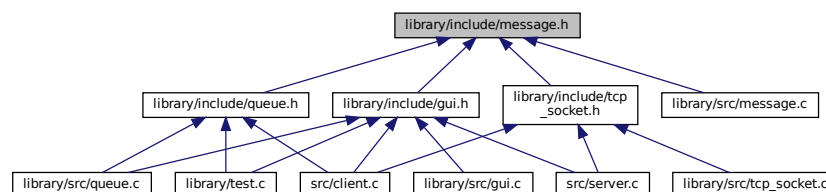
Returns

THLNode*

5.8 library/include/message.h File Reference

Definition of struct Message.

This graph shows which files directly or indirectly include this file:



Classes

- struct [SMessage](#)

Macros

- `#define USR_MAXLEN 64`
- `#define TXT_MAXLEN 1024`
- `#define MSG_MAXLEN 32`
- `#define MESSAGE 0`
Message Codes for protocol.
- `#define MSG_CLI_CREATE 1`
- `#define MSG_SRV_USRACK 2`
- `#define MSG_SRV_USRNCK 3`
- `#define MSG_SRV_USRNRC 4`
- `#define MSG_CLI_LSTUSR 5`
- `#define MSG_SRV_AVLUSR 6`
- `#define MESSAGE_STRUCT`
struct exchanged by parties for gagchat

Typedefs

- `typedef struct SMessage Message`

Functions

- `Message * message_create ()`
Creates an empty message.
- `void message_constructor (Message *message, char *user, char *text)`
Constructs a basic exchangeable message.
- `void message_code_constructor (Message *message, char *user, char *text, int code)`
Constructs a code message.
- `void message_destroy (Message *message)`
Destroys a Message.

5.8.1 Detailed Description

Definition of struct Message.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-28

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.8.2 Macro Definition Documentation

5.8.2.1 MESSAGE

```
#define MESSAGE 0
```

Message Codes for protocol.

5.8.2.2 MESSAGE_STRUCT

```
#define MESSAGE_STRUCT
```

struct exchanged by parties for gagchat

5.8.3 Function Documentation

5.8.3.1 message_code_constructor()

```
void message_code_constructor (
    Message * message,
    char * user,
    char * text,
    int code )
```

Constructs a code message.

Parameters

| | |
|-------------|----------------|
| <i>user</i> | Username |
| <i>text</i> | Message text |
| <i>code</i> | code to append |

Returns

void

5.8.3.2 message_constructor()

```
void message_constructor (
    Message * message,
```

```
char * user,  
char * text )
```

Constructs a basic exchangeable message.

Parameters

| | |
|-------------|--------------|
| <i>user</i> | Username |
| <i>text</i> | Message text |

Returns

void

5.8.3.3 message_create()

```
Message* message_create ( )
```

Creates an empty message.

Returns

Message* pointer to the Empty message.

5.8.3.4 message_destroy()

```
void message_destroy (  
    Message * message )
```

Destroys a Message.

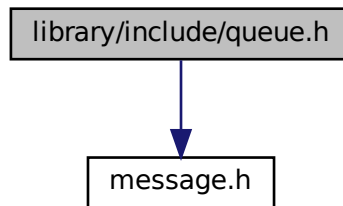
Parameters

| | |
|----------------|-----------------------------------|
| <i>message</i> | the Message reference to destroy. |
|----------------|-----------------------------------|

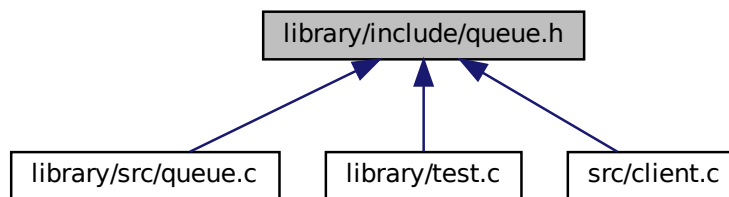
5.9 library/include/queue.h File Reference

```
#include "message.h"
```

Include dependency graph for queue.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct `TQueue`

Struct of the queue.

Functions

- `TQueue * queueCreate ()`
- `void queueDestroy (TQueue *q)`
- `void queueAdd (TQueue *q, Message x)`
- `Message * queueRemove (TQueue *q)`
- `Message * queueFront (TQueue *q)`
- `int queueIsEmpty (TQueue *q)`
- `void queuePrint (TQueue *q, char *user_sender)`

5.9.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-11-03

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

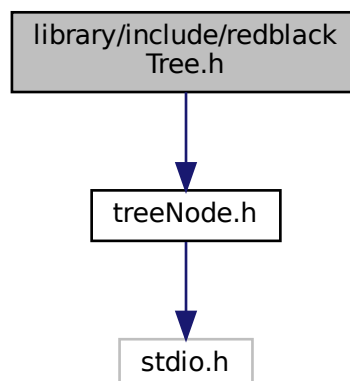
gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

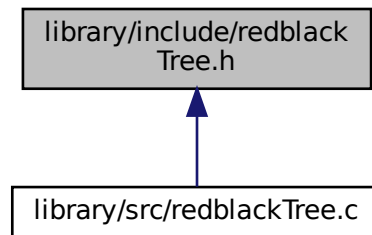
5.10 library/include/redblackTree.h File Reference

```
#include "treeNode.h"
```

Include dependency graph for redblackTree.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [RBTREE](#)

Functions

- [RBTREE](#) * **rbtree_create** (int(*comparator)(const void *, const void *))
- void **rbtree_insert** ([RBTREE](#) *rbtree, void *data)
- void **rbtree_deleteData** ([RBTREE](#) *rbtree, void *data)
- void **rbtree_destroy** ([RBTREE](#) *rbtree)

5.10.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

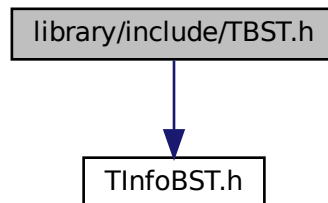
gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

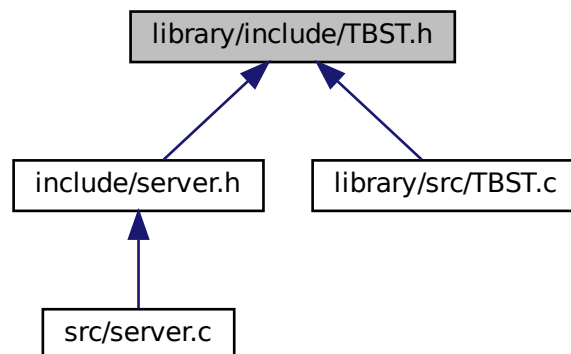
5.11 library/include/TBST.h File Reference

```
#include "TInfoBST.h"
```

Include dependency graph for TBST.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [SBSTnode](#)

Typedefs

- typedef struct [SBSTnode](#) **TBSTnode**
- typedef [TBSTnode](#) * **TBST**

Functions

- [TBSTnode](#) * **BSTnodeCreate** (TInfoBST)
- void **BSTnodeDestroy** ([TBSTnode](#) *)
- [TBST](#) **BSTcreate** ()
- [TBST](#) **BSTdestroy** ([TBST](#))
- [TBST](#) **BSTsearch** ([TBST](#), TInfoBST)
- [TBST](#) **BSTmin** ([TBST](#))
- [TBST](#) **BSTmax** ([TBST](#))
- [TBST](#) **BSTinsert** ([TBST](#), TInfoBST)
- [TBST](#) **BSTdelete** ([TBST](#), TInfoBST)
- [TBST](#) **BSTsearchI** ([TBST](#), TInfoBST)
- [TBST](#) **BSTinsertI** ([TBST](#), TInfoBST)
- [TBST](#) **BSTdeleteI** ([TBST](#), TInfoBST)
- void **BSTprint** ([TBST](#))
- void **BSTprintStructure** ([TBST](#))

5.11.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

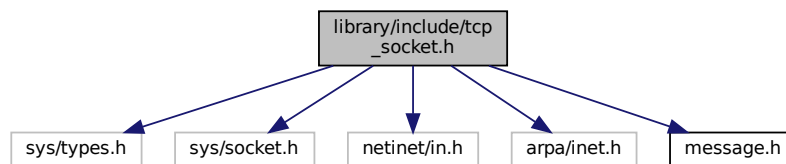
You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.12 library/include/tcp_socket.h File Reference

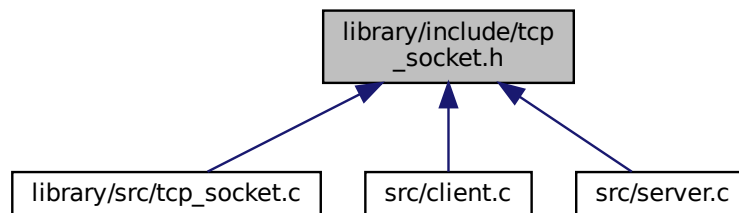
Implementation of tcp socket both for client and server.

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include "message.h"
```

Include dependency graph for tcp_socket.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [STCPSocket](#)

Macros

- #define **SERV_PORT** 3000
- #define **LISTENQ** 100

Typedefs

- typedef enum EConnType **ConnType**
- typedef struct [STCPSocket](#) **TCPSocket**

Enumerations

- enum **EConnType** { **CLIENT**, **SERVER** }

Functions

- **TCPSocket** * [tcp_socket_create](#) (ConnType type, char *server_ip)
Creates a TCPSocket and pass its pointer.
- void [tcp_socket_server_listen](#) (**TCPSocket** *tcp_socket)
Configures the server in listening mode.
- int [tcp_socket_server_accept](#) (**TCPSocket** *tcp_socket)
Accepts a connection from a requesting client.
- void [tcp_socket_client_connect](#) (**TCPSocket** *tcp_socket)
Connects the client to the specified TCPSocket.
- int [tcp_socket_send_message](#) (int connection_fd, **Message** *message)
Sends a Message on TCPSocket.
- int [tcp_socket_recv_message](#) (int connection_fd, **Message** *message)
Receives a Message from TCPSocket.
- void [tcp_socket_destroy](#) (**TCPSocket** *tcp_socket)
Destroys a TCPSocket.

5.12.1 Detailed Description

Implementation of tcp socket both for client and server.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.12.2 Function Documentation

5.12.2.1 tcp_socket_client_connect()

```
void tcp_socket_client_connect (
    TCPSocket * tcp_socket )
```

Connects the client to the specified TCPSocket.

Parameters

| | |
|-------------------|------------------------------|
| <i>tcp_socket</i> | pointer to TCPSocket struct. |
|-------------------|------------------------------|

5.12.2.2 tcp_socket_create()

```
TCPSocket* tcp_socket_create (
    ConnType type,
    char * server_ip )
```

Creates a TCPSocket and pass its pointer.

Parameters

| | |
|------------------------|--|
| <i>type</i> | the TCPSocket type (client or server) |
| <i>server↔ _ip</i> | if type = CLIENT, is mandatory to specify the server's ip address formatted as string. |

Returns

TCPSocket* pointer to TCPSocket struct

5.12.2.3 tcp_socket_destroy()

```
void tcp_socket_destroy (
    TCPSocket * tcp_socket )
```

Destroys a TCPSocket.

Parameters

| | |
|-------------------|-------------------------------------|
| <i>tcp_socket</i> | the TCPSocket reference to destroy. |
|-------------------|-------------------------------------|

5.12.2.4 tcp_socket_recv_message()

```
int tcp_socket_recv_message (
    int connection_fd,
    Message * message )
```

Receives a Message from TCPSocket.

Parameters

| | |
|----------------------|---|
| <i>connection_fd</i> | connection file descriptor. If client, it is param sockfd of TCPSocket. If server, it is obtained by the output of tcp_socket_server_accept function. |
| <i>message</i> | pointer to the message to receive. |

Returns

int returns the number of bytes received, or -1 if an error occurred. In the event of an error, errno is set to indicate the error.

5.12.2.5 tcp_socket_send_message()

```
int tcp_socket_send_message (
    int connection_fd,
    Message * message )
```

Sends a Message on TCPSocket.

Parameters

| | |
|----------------------|---|
| <i>connection_fd</i> | connection file descriptor. If client, it is param sockfd of TCPSocket. If server, it is obtained by the output of tcp_socket_server_accept function. |
| <i>message</i> | the Message to send. |

Returns

int On success, returns the number of bytes sent. On error, -1 is returned, and errno is set to indicate the error.

5.12.2.6 tcp_socket_server_accept()

```
int tcp_socket_server_accept (
    TCPSocket * tcp_socket )
```

Accepts a connection from a requesting client.

Parameters

| | |
|-------------------------|-------------------------------|
| <code>tcp_socket</code> | pointer to TCP Socket struct. |
|-------------------------|-------------------------------|

Returns

int the connection file descriptor.

5.12.2.7 tcp_socket_server_listen()

```
void tcp_socket_server_listen (  
    TCP Socket * tcp_socket )
```

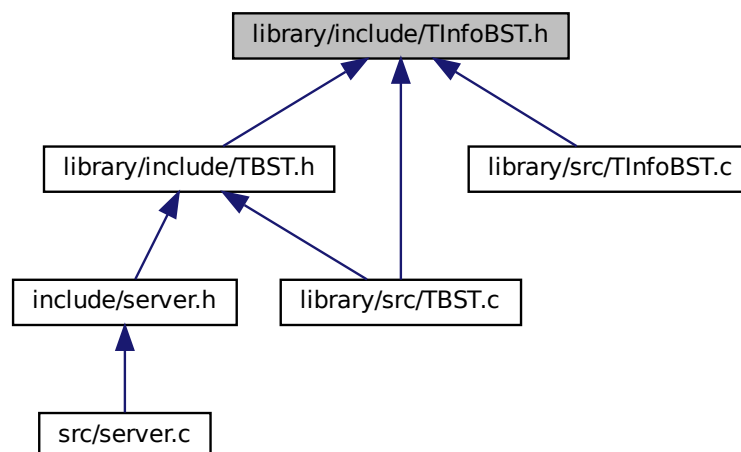
Configures the server in listening mode.

Parameters

| | |
|-------------------------|-------------------------------|
| <code>tcp_socket</code> | pointer to TCP Socket struct. |
|-------------------------|-------------------------------|

5.13 library/include/TInfoBST.h File Reference

This graph shows which files directly or indirectly include this file:

**Typedefs**

- typedef char * **TInfoBST**

Functions

- int **infoBSTEqual** (TInfoBST, TInfoBST)
- int **infoBSTGreater** (TInfoBST, TInfoBST)
- int **infoBSTLess** (TInfoBST, TInfoBST)
- void **infoBSTPrint** (TInfoBST)

5.13.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

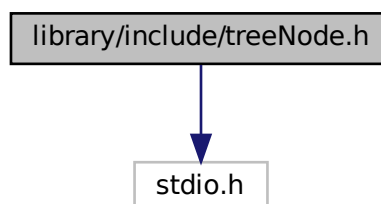
gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

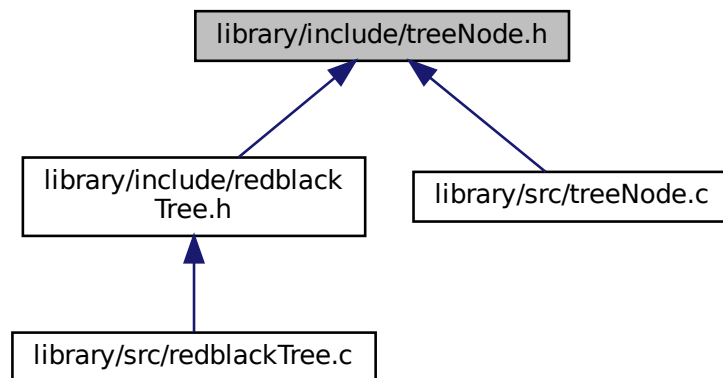
5.14 library/include/treeNode.h File Reference

```
#include <stdio.h>
```

Include dependency graph for treeNode.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [Treenode](#)

Typedefs

- typedef struct [Treenode](#) [Treenode](#)

Functions

- [Treenode](#) * [treenode_createGuard](#) ()
- void [treenode_LeftRotate](#) ([Treenode](#) **root, [Treenode](#) *x, [Treenode](#) *guard)
- void [treenode_RightRotate](#) ([Treenode](#) **root, [Treenode](#) *y, [Treenode](#) *guard)
- void [treenode_insertion](#) ([Treenode](#) **root, void *data, int(*comparator)(const void *, const void *), [Treenode](#) *guard)
- void [treenode_Fixedinsertion](#) ([Treenode](#) **root, [Treenode](#) *z, [Treenode](#) *guard)
- void [treenode_deleteTree](#) ([Treenode](#) *root, [Treenode](#) *guard)
- [Treenode](#) * [treenode_FindData](#) ([Treenode](#) *root, void *data, int(*comparator)(const void *, const void *), [Treenode](#) *guard)
- void [treenode_DeleteData](#) ([Treenode](#) **root, void *data, int(*comparator)(const void *, const void *), [Treenode](#) *guard)
- void [treenode_FixedDelete](#) ([Treenode](#) **root, [Treenode](#) *x, [Treenode](#) *guard)
- void [treenode_transplant](#) ([Treenode](#) **root, [Treenode](#) *x, [Treenode](#) *y, [Treenode](#) *guard)

5.14.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

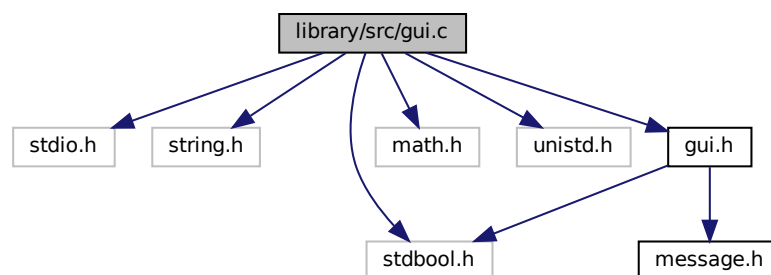
This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.15 library/src/gui.c File Reference

```
#include <stdio.h>
#include <string.h>
#include <stdbool.h>
#include <math.h>
#include <unistd.h>
#include "gui.h"
Include dependency graph for gui.c:
```



Functions

- void `gui_set_color` (char *color)
set the color of text or background, apply a text style
- void `gui_print_n_times` (char *str, int n)
Print.
- void `gui_print_horizontal_line` ()
Print the upper and lower separator of box.
- void `gui_print_horizontal_msg_line` (bool is_sender)
Print the upper and lower separator of the text box.
- void `gui_print_user_msg` (char *user, bool is_sender)
- void `gui_print_msg` (char *msg, bool is_sender)
- void `gui_print_message` (Message *msg, bool is_sender)
- void `gui_clear_Screen` ()
- void `gui_print_line` (char *line)
- void `gui_print_menu` (char *user)
- void `gui_print_list_users_header` ()
- void `gui_print_list_users_footer` ()
- void `gui_print_list_user` (char *user)

5.15.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-11-03

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.15.2 Function Documentation

5.15.2.1 `gui_print_horizontal_line()`

```
void gui_print_horizontal_line ( )
```

Print the upper and lower separator of box.

5.15.2.2 `gui_print_horizontal_msg_line()`

```
void gui_print_horizontal_msg_line (
    bool is_sender )
```

Print the upper and lower separator of the text box.

Parameters

| | |
|------------------|---|
| <i>is_sender</i> | true if is a message received, else false |
|------------------|---|

5.15.2.3 `gui_print_message()`

```
void gui_print_message (
    Message * msg,
    bool received )
```

Parameters

| | |
|-----------------|--|
| <i>msg</i> | |
| <i>received</i> | |

5.15.2.4 `gui_print_n_times()`

```
void gui_print_n_times (
    char * str,
    int n )
```

Print.

Parameters

| | |
|------------|-----------------|
| <i>n</i> | time the string |
| <i>str</i> | |
| <i>str</i> | string to print |
| <i>n</i> | times |

5.15.2.5 `gui_print_user_msg()`

```
void gui_print_user_msg (
    char * user,
    bool is_sender )
```

Parameters

| | |
|------------------|---|
| <i>user</i> | |
| <i>is_sender</i> | true if is a message received, else false |

5.15.2.6 `gui_set_color()`

```
void gui_set_color (
    char * color )
```

set the color of text or background, apply a text style

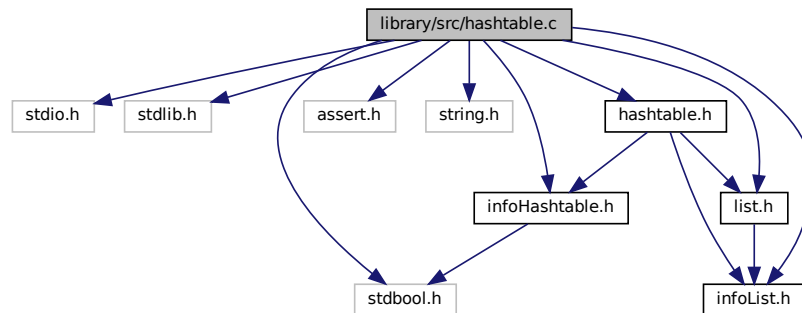
Parameters

| | |
|--------------|--|
| <i>color</i> | |
|--------------|--|

5.16 `library/src/hashtable.c` File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <stdbool.h>
#include <assert.h>
#include <string.h>
#include "infoList.h"
#include "list.h"
#include "hashtable.h"
#include "infoHashtable.h"
```

Include dependency graph for hashtable.c:



Functions

- `THashTable *` `hashTableCreate` ()
Creation of a new HashTable.
- `void` `hashTableResize` (`THashTable` *ht, int n)
resizing Hashtable
- `void` `hashTableDestroy` (`THashTable` *ht)
destroy the HashTable
- `TValueHashtable` `hashTableSearch` (`THashTable` *ht, `TKeyHashtable` key)
search a key in a HashTable, return the list
- `void` `hashTableInsert` (`THashTable` *ht, `TKeyHashtable` key, `TValueHashtable` value)
Insert a new key and value in a HashTable.
- `void` `hashTableDelete` (`THashTable` *ht, `TKeyHashtable` key)
Delete a key in a HashTable.
- `void` `hashTablePrint` (`THashTable` *ht)
Print a HashTable.
- `u_int64_t` `keyHash` (`TKeyHashtable` key)
hash a key (function h(x))
- `u_int64_t` `keyHashD` (`TKeyHashtable` key)
hash a key (function d(x))
- `u_int64_t` `keyHashExpanse` (`TKeyHashtable` key, `u_int64_t` hash, int j)
*double hashing return $hash = h(x) + j * d(x)$*

5.16.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.16.2 Function Documentation

5.16.2.1 hashTableCreate()

```
THashTable* hashTableCreate ( )
```

Creation of a new HashTable.

Returns

THashTable*

5.16.2.2 hashTableDelete()

```
void hashTableDelete (
    THashTable * ht,
    TKeyHashtable key )
```

Delete a key in a HashTable.

Parameters

| | |
|------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to delete |

5.16.2.3 `hashTableDestroy()`

```
void hashTableDestroy (
    THashTable * ht )
```

destroy the HashTable

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
|-----------|----------------------|

5.16.2.4 `hashTableInsert()`

```
void hashTableInsert (
    THashTable * ht,
    TKeyHashtable key,
    TValueHashtable value )
```

Insert a new key and value in a HashTable.

Parameters

| | |
|--------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to insert |
| <i>value</i> | value to insert |

5.16.2.5 `hashTablePrint()`

```
void hashTablePrint (
    THashTable * ht )
```

Print a HashTable.

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
|-----------|----------------------|

5.16.2.6 `hashTableResize()`

```
void hashTableResize (
    THashTable * ht,
    int n )
```

resizing Hashtable

Resize hashtable.

Parameters

| | |
|-----------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>n</i> | new size |

5.16.2.7 hashTableSearch()

```
TValueHashtable hashTableSearch (
    THashTable * ht,
    TKeyHashtable key )
```

search a key in a HashTable, return the list

search in a HashTable

Parameters

| | |
|------------|----------------------|
| <i>ht</i> | pointer of HashTable |
| <i>key</i> | key to find |

Returns

TValueHashtable

5.16.2.8 keyHash()

```
u_int64_t keyHash (
    TKeyHashtable key )
```

hash a key (function h(x))

Returns

u_int64_t

5.16.2.9 keyHashD()

```
u_int64_t keyHashD (
    TKeyHashtable key )
```

hash a key (function d(x))

Returns

u_int64_t

5.16.2.10 keyHashExpand()

```
u_int64_t keyHashExpand (
    TKeyHashtable key,
    u_int64_t hash,
    int j )
```

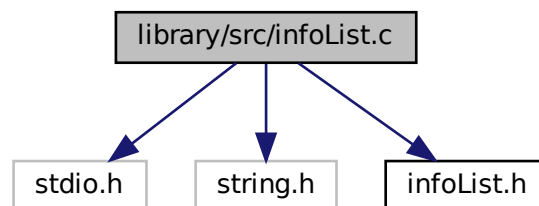
double hashing return $hash = h(x) + j*d(x)$

Returns

u_int64_t

5.17 library/src/infoList.c File Reference

```
#include <stdio.h>
#include <string.h>
#include "infoList.h"
Include dependency graph for infoList.c:
```



Functions

- int `infoListEqual` (TInfoList info1, TInfoList info2)
check if all the information of frist value are the same of the second value
- void `infoListPrint` (TInfoList info)
structured print of the value

5.17.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.17.2 Function Documentation

5.17.2.1 infoListEqual()

```
int infoListEqual (
    TInfoList info1,
    TInfoList info2 )
```

check if all the information of frist value are the same of the second value

Returns

bool

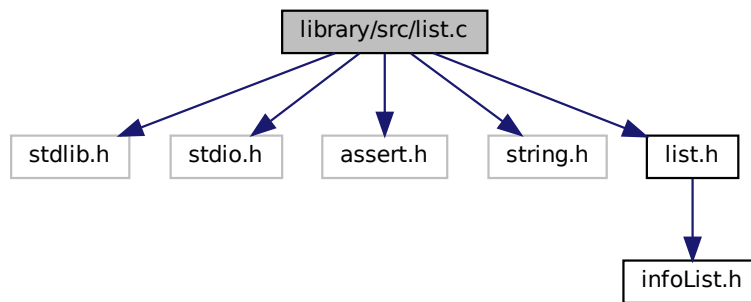
5.17.2.2 infoListPrint()

```
void infoListPrint (
    TInfoList info )
```

structured print of the value

5.18 library/src/list.c File Reference

```
#include <stdlib.h>
#include <stdio.h>
#include <assert.h>
#include <string.h>
#include "list.h"
Include dependency graph for list.c:
```



Functions

- `THLNode * nodeCreate (TInfoList info)`
- `void nodeDestroy (THLNode *node)`
- `TList listCreate ()`
Creation of a new List.
- `TList listDestroy (TList list)`
Destroy a List.
- `TList listInsert (TList l, TInfoList info)`
Insert a new value in the List.
- `THLNode * listSearch (TList list, TInfoList info)`
Search an element into the List.
- `TList listDelete (TList list, TInfoList info)`
Delete an element from a list.
- `void listPrint (TList list)`
Print of an entire List.

5.18.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.18.2 Function Documentation

5.18.2.1 listCreate()

```
TList listCreate ( )
```

Creation of a new List.

Returns

TList

5.18.2.2 listDelete()

```
TList listDelete (
    TList list,
    TInfoList info )
```

Delete an element from a list.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

Returns

TList

5.18.2.3 listDestroy()

```
TList listDestroy (  
    TList list )
```

Destroy a List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
|-------------|-----------------|

Returns

TList

5.18.2.4 listInsert()

```
TList listInsert (  
    TList l,  
    TInfoList info )
```

Insert a new value in the List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

Returns

TList

5.18.2.5 listPrint()

```
void listPrint (
    TList list )
```

Print of an entire List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
|-------------|-----------------|

5.18.2.6 listSearch()

```
THLNode* listSearch (
    TList list,
    TInfoList info )
```

Search an element into the List.

Parameters

| | |
|-------------|-----------------|
| <i>list</i> | pointer of List |
| <i>info</i> | value of List |

Returns

THLNode*

5.18.2.7 nodeCreate()

```
THLNode* nodeCreate (
    TInfoList info )
```

Parameters

| | |
|-------------|--|
| <i>info</i> | |
|-------------|--|

Returns

THLNode*

5.18.2.8 nodeDestroy()

```
void nodeDestroy (
    THLNode * node )
```

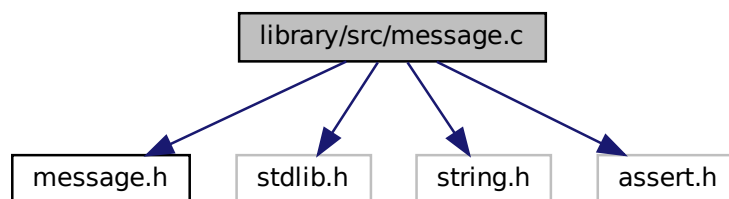
Parameters

| | |
|-------------|--|
| <i>node</i> | |
|-------------|--|

5.19 library/src/message.c File Reference

```
#include "message.h"
#include <stdlib.h>
#include <string.h>
#include <assert.h>
```

Include dependency graph for message.c:



Functions

- `Message * message_create ()`
Creates an empty message.
- `void message_constructor (Message *msg, char *user, char *text)`
Constructs a basic exchangeable message.
- `void message_code_constructor (Message *msg, char *user, char *text, int code)`
Constructs a code message.
- `void message_destroy (Message *message)`
Destroys a Message.

5.19.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.19.2 Function Documentation

5.19.2.1 `message_code_constructor()`

```
void message_code_constructor (
    Message * message,
    char * user,
    char * text,
    int code )
```

Constructs a code message.

Parameters

| | |
|-------------|----------------|
| <i>user</i> | Username |
| <i>text</i> | Message text |
| <i>code</i> | code to append |

Returns

void

5.19.2.2 message_constructor()

```
void message_constructor (
    Message * message,
    char * user,
    char * text )
```

Constructs a basic exchangeable message.

Parameters

| | |
|-------------|--------------|
| <i>user</i> | Username |
| <i>text</i> | Message text |

Returns

void

5.19.2.3 message_create()

```
Message* message_create ( )
```

Creates an empty message.

Returns

Message* pointer to the Empty message.

5.19.2.4 message_destroy()

```
void message_destroy (
    Message * message )
```

Destroys a Message.

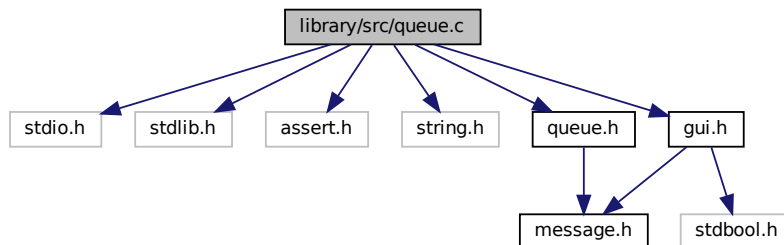
Parameters

| | |
|----------------|-----------------------------------|
| <i>message</i> | the Message reference to destroy. |
|----------------|-----------------------------------|

5.20 library/src/queue.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
```

```
#include <assert.h>
#include <string.h>
#include "queue.h"
#include "gui.h"
Include dependency graph for queue.c:
```



Functions

- `TQueue * queueCreate (int queue_size)`
- `void queueDestroy (TQueue *q)`
- `void queueAdd (TQueue *q, Message x)`
- `Message * queueRemove (TQueue *q)`
- `Message * queueFront (TQueue *q)`
- `int queueIsEmpty (TQueue *q)`
- `void queuePrint (TQueue *q, char *user_sender)`

5.20.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-11-03

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

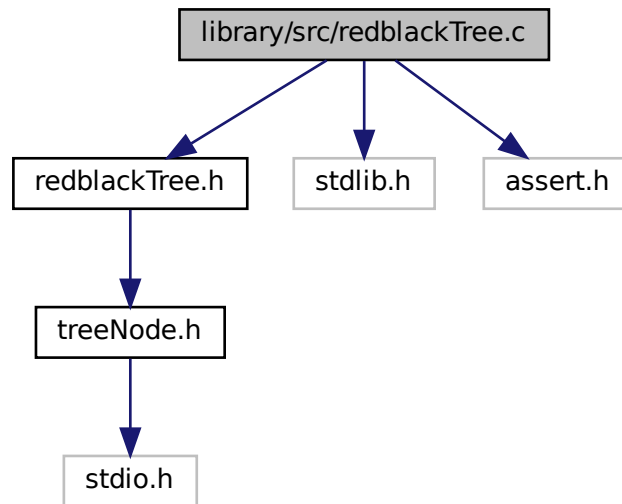
5.21 library/src/redblackTree.c File Reference

```
#include "redblackTree.h"
```

```
#include <stdlib.h>
```

```
#include <assert.h>
```

Include dependency graph for redblackTree.c:



Functions

- `RBTree * rbtree_create (int(*comparator)(const void *, const void *))`
- `void rbtree_insert (RBTree *rbtree, void *data)`
- `void rbtree_deleteData (RBTree *rbtree, void *data)`
- `void rbtree_destroy (RBTree *rbtree)`

5.21.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

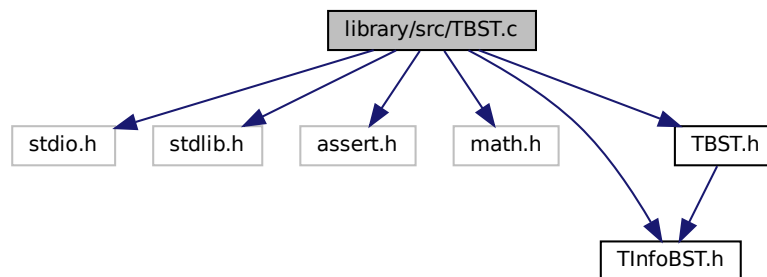
This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.22 library/src/TBST.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <assert.h>
#include <math.h>
#include "TInfoBST.h"
#include "TBST.h"
Include dependency graph for TBST.c:
```



Functions

- **TBSTnode** * **BSTnodeCreate** (TInfoBST info)
- void **BSTnodeDestroy** (TBSTnode *node)
- **TBST** **BSTcreate** ()
- **TBST** **BSTdestroy** (TBST tree)
- **TBST** **BSTsearch** (TBST tree, TInfoBST info)
- **TBST** **BSTmin** (TBST tree)
- **TBST** **BSTmax** (TBST tree)
- **TBST** **BSTinsert** (TBST tree, TInfoBST info)
- **TBST** **BSTdelete** (TBST tree, TInfoBST info)
- void **BSTprint** (TBST tree)
- **TBST** **BSTsearchI** (TBST tree, TInfoBST info)
- **TBST** **BSTinsertI** (TBST tree, TInfoBST info)
- **TBST** **BSTdeleteI** (TBST tree, TInfoBST info)
- int **BSTdepth** (TBST tree)
- void **BSTprintStructure1** (TBST tree, int o, TInfoBST *n, int *m)

5.22.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

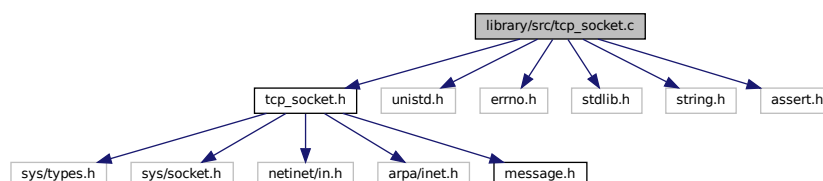
This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.23 library/src/tcp_socket.c File Reference

```
#include "tcp_socket.h"
#include <unistd.h>
#include <errno.h>
#include <stdlib.h>
#include <string.h>
#include <assert.h>
Include dependency graph for tcp_socket.c:
```



Functions

- `TCPSocket * tcp_socket_create` (ConnType type, char *server_ip)
Creates a TCPSocket and pass its pointer.
- void `tcp_socket_server_listen` (TCPSocket *tcp_socket)
Configures the server in listening mode.
- int `tcp_socket_server_accept` (TCPSocket *tcp_socket)
Accepts a connection from a requesting client.
- void `tcp_socket_client_connect` (TCPSocket *tcp_socket)
Connects the client to the specified TCPSocket.
- int `tcp_socket_send_message` (int connection_fd, Message *message)
Sends a Message on TCPSocket.
- int `tcp_socket_recv_message` (int connection_fd, Message *message)
Receives a Message from TCPSocket.
- void `tcp_socket_destroy` (TCPSocket *tcp_socket)
Destroys a TCPSocket.

5.23.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-30

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.23.2 Function Documentation

5.23.2.1 tcp_socket_client_connect()

```
void tcp_socket_client_connect (  
    TCPSocket * tcp_socket )
```

Connects the client to the specified TCPSocket.

Parameters

| | |
|-------------------|------------------------------|
| <i>tcp_socket</i> | pointer to TCPSocket struct. |
|-------------------|------------------------------|

5.23.2.2 tcp_socket_create()

```
TCPSocket* tcp_socket_create (
    ConnType type,
    char * server_ip )
```

Creates a TCPSocket and pass its pointer.

Parameters

| | |
|------------------|--|
| <i>type</i> | the TCPSocket type (client or server) |
| <i>server_ip</i> | if type = CLIENT, is mandatory to specify the server's ip address formatted as string. |

Returns

TCPSocket* pointer to TCPSocket struct

5.23.2.3 tcp_socket_destroy()

```
void tcp_socket_destroy (
    TCPSocket * tcp_socket )
```

Destroys a TCPSocket.

Parameters

| | |
|-------------------|-------------------------------------|
| <i>tcp_socket</i> | the TCPSocket reference to destroy. |
|-------------------|-------------------------------------|

5.23.2.4 tcp_socket_recv_message()

```
int tcp_socket_recv_message (
    int connection_fd,
    Message * message )
```

Receives a Message from TCPSocket.

Parameters

| | |
|----------------------------|---|
| <i>connection↔ _fd</i> | connection file descriptor. If client, it is param sockfd of TCPSocket. If server, it is obtained by the output of tcp_socket_server_accept function. |
| <i>message</i> | pointer to the message to receive. |

Returns

int returns the number of bytes received, or -1 if an error occurred. In the event of an error, errno is set to indicate the error.

5.23.2.5 tcp_socket_send_message()

```
int tcp_socket_send_message (
    int connection_fd,
    Message * message )
```

Sends a Message on TCPSocket.

Parameters

| | |
|----------------------------|---|
| <i>connection↔ _fd</i> | connection file descriptor. If client, it is param sockfd of TCPSocket. If server, it is obtained by the output of tcp_socket_server_accept function. |
| <i>message</i> | the Message to send. |

Returns

int On success, returns the number of bytes sent. On error, -1 is returned, and errno is set to indicate the error.

5.23.2.6 tcp_socket_server_accept()

```
int tcp_socket_server_accept (
    TCPSocket * tcp_socket )
```

Accepts a connection from a requesting client.

Parameters

| | |
|-------------------|------------------------------|
| <i>tcp_socket</i> | pointer to TCPSocket struct. |
|-------------------|------------------------------|

Returns

int the connection file descriptor.

5.23.2.7 tcp_socket_server_listen()

```
void tcp_socket_server_listen (
    TCPSocket * tcp_socket )
```

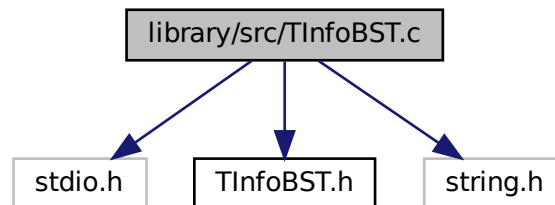
Configures the server in listening mode.

Parameters

| | |
|-------------------|------------------------------|
| <i>tcp_socket</i> | pointer to TCPSocket struct. |
|-------------------|------------------------------|

5.24 library/src/TInfoBST.c File Reference

```
#include <stdio.h>
#include "TInfoBST.h"
#include <string.h>
Include dependency graph for TInfoBST.c:
```



Functions

- int **infoBSTEqual** (TInfoBST info1, TInfoBST info2)
- int **infoBSTGreater** (TInfoBST info1, TInfoBST info2)
- int **infoBSTLess** (TInfoBST info1, TInfoBST info2)
- void **infoBSTPrint** (TInfoBST info)

5.24.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

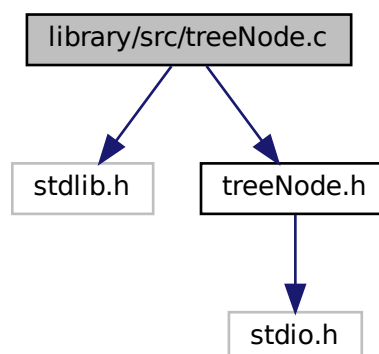
This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.25 library/src/treeNode.c File Reference

```
#include <stdlib.h>
#include "treeNode.h"
Include dependency graph for treeNode.c:
```



Functions

- void **treenode_LeftRotate** (**Treenode** **root, **Treenode** *x, **Treenode** *guard)
- void **treenode_RightRotate** (**Treenode** **root, **Treenode** *x, **Treenode** *guard)
- void **treenode_insertion** (**Treenode** **root, void *data, int(*comparator)(const void *, const void *), **Treenode** *guard)
- void **treenode_Fixedinsertion** (**Treenode** **root, **Treenode** *z, **Treenode** *guard)
- **Treenode** * **treenode_FindData** (**Treenode** *root, void *data, int(*comparator)(const void *, const void *), **Treenode** *guard)
- void **treenode_DeleteData** (**Treenode** **root, void *data, int(*comparator)(const void *, const void *), **Treenode** *guard)
- void **treenode_transplant** (**Treenode** **root, **Treenode** *x, **Treenode** *y, **Treenode** *guard)
- void **treenode_FixedDelete** (**Treenode** **root, **Treenode** *x, **Treenode** *guard)
- void **treenode_deleteTree** (**Treenode** *root, **Treenode** *guard)
- void **treenode_printTree** (FILE *f, **Treenode** *root, **Treenode** *guard)
- **Treenode** * **treenode_createGuard** ()

5.25.1 Detailed Description

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-11-04

Copyright

Copyright (c) 2022 - All Rights Reserved

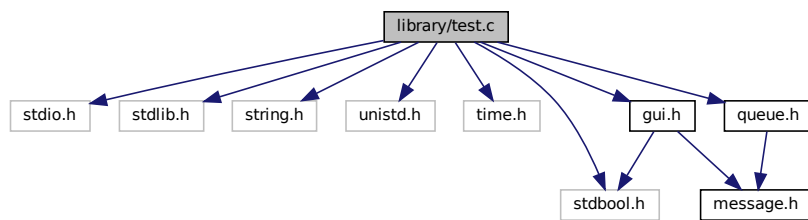
This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.26 library/test.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
#include <time.h>
#include <stdbool.h>
#include "gui.h"
#include "queue.h"
Include dependency graph for test.c:
```



Functions

- `int main (int argc, char **argv)`

5.26.1 Detailed Description

Author

Gianluca (g.canzolino3@studenti.unisa.it)

Version

0.1

Date

2022-10-29

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

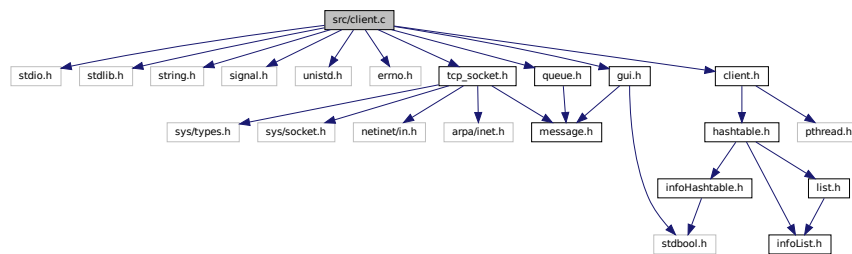
You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.27 src/client.c File Reference

Client Implementation of gagchat.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <signal.h>
#include <unistd.h>
#include <errno.h>
#include "tcp_socket.h"
#include "client.h"
#include "queue.h"
#include "gui.h"
```

Include dependency graph for client.c:



Functions

- void * [client_message_receiver](#) (void *arg)
thread for message receiving.
- int **main** (int argc, char **argv)

Variables

- [TCPSocket](#) * **sock**

5.27.1 Detailed Description

Client Implementation of gagchat.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-28

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.27.2 Function Documentation**5.27.2.1 client_message_receiver()**

```
void* client_message_receiver (
    void * arg )
```

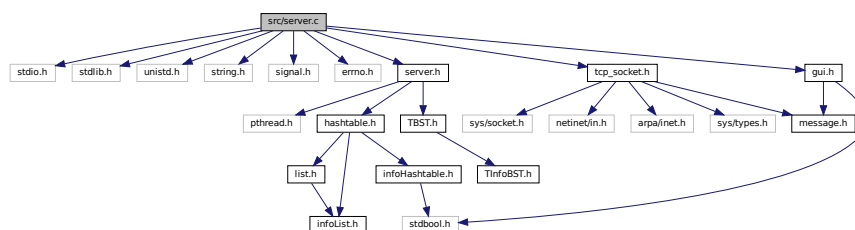
thread for message receiving.

5.28 src/server.c File Reference

Server implementation of gagchat.

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <signal.h>
#include <errno.h>
#include "server.h"
#include "tcp_socket.h"
#include "gui.h"
```

Include dependency graph for server.c:



Functions

- void `server_sendToAll` (`THashTable` *ht, `TBST` tree, `Message` *message)
Sends a message to all clients.
- void `server_sendUsersTo` (int connection_fd, `TBST` users)
Sends the users list to specified client.
- void * `server_manage_client` (void *arg)
Thread for managing a generic client.
- int `main` (int argc, char **argv)

Variables

- `TCPSocket` * `sock`

5.28.1 Detailed Description

Server implementation of gagchat.

Author

Giuseppe Gambardella (g.gambardella23@studenti.unisa.it)

Version

0.1

Date

2022-10-30

Copyright

Copyright (c) 2022 - All Rights Reserved

This file is part of gagchat. gagchat is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

gagchat is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with gagchat. If not, see <http://www.gnu.org/licenses/>.

5.28.2 Function Documentation

5.28.2.1 server_manage_client()

```
void* server_manage_client (
    void * arg )
```

Thread for managing a generic client.

5.28.2.2 server_sendToAll()

```
void server_sendToAll (
    THashTable * ht,
    TBST tree,
    Message * message )
```

Sends a message to all clients.

Parameters

| | |
|----------------|---------------------------------------|
| <i>ht</i> | the hashtable containing connections. |
| <i>tree</i> | tree list of users. |
| <i>message</i> | the message to send. |

5.28.2.3 server_sendUsersTo()

```
void server_sendUsersTo (
    int connection_fd,
    TBST users )
```

Sends the users list to specified client.

Parameters

| | |
|-----------------------------------|-----------------------------|
| <i>connection</i> ↔ <i>_fd</i> | connection file descriptor. |
| <i>users</i> | tree with all of users. |

Index

- client.c
 - client_message_receiver, [84](#)
- client.h
 - client_message_receiver, [17](#)
- client_message_receiver
 - client.c, [84](#)
 - client.h, [17](#)
- gui.c
 - gui_print_horizontal_line, [56](#)
 - gui_print_horizontal_msg_line, [57](#)
 - gui_print_message, [57](#)
 - gui_print_n_times, [57](#)
 - gui_print_user_msg, [58](#)
 - gui_set_color, [58](#)
- gui.h
 - gui_print_message, [23](#)
 - gui_set_color, [23](#)
- gui_print_horizontal_line
 - gui.c, [56](#)
- gui_print_horizontal_msg_line
 - gui.c, [57](#)
- gui_print_message
 - gui.c, [57](#)
 - gui.h, [23](#)
- gui_print_n_times
 - gui.c, [57](#)
- gui_print_user_msg
 - gui.c, [58](#)
- gui_set_color
 - gui.c, [58](#)
 - gui.h, [23](#)
- hashtable.c
 - hashTableCreate, [60](#)
 - hashTableDelete, [60](#)
 - hashTableDestroy, [60](#)
 - hashTableInsert, [61](#)
 - hashTablePrint, [61](#)
 - hashTableResize, [61](#)
 - hashTableSearch, [62](#)
 - keyHash, [62](#)
 - keyHashD, [62](#)
 - keyHashExpand, [63](#)
- hashtable.h
 - hashTableCreate, [26](#)
 - hashTableDelete, [26](#)
 - hashTableDestroy, [27](#)
 - hashTableInsert, [27](#)
 - hashTablePrint, [27](#)
 - hashTableResize, [28](#)
 - hashTableSearch, [28](#)
 - keyHash, [29](#)
 - keyHashD, [29](#)
 - keyHashExpand, [29](#)
 - TBucket, [26](#)
- include/client.h, [15](#)
- include/server.h, [17](#)
- infoEqual
 - infoHashtable.h, [32](#)
- infoGreater
 - infoHashtable.h, [32](#)
- infoHashtable.h
 - infoEqual, [32](#)
 - infoGreater, [32](#)
 - infoLess, [32](#)
 - infoPrint, [32](#)
 - TKeyHashtable, [31](#)
- infoLess
 - infoHashtable.h, [32](#)
- infoList.c
 - infoListEqual, [64](#)
 - infoListPrint, [64](#)
- infoList.h
 - infoListEqual, [34](#)
 - infoListPrint, [35](#)
 - TInfoList, [34](#)
- infoListEqual

- infoList.c, 64
- infoList.h, 34
- infoListPrint
 - infoList.c, 64
 - infoList.h, 35
- infoPrint
 - infoHashtable.h, 32
- keyHash
 - hashtable.c, 62
 - hashtable.h, 29
- keyHashD
 - hashtable.c, 62
 - hashtable.h, 29
- keyHashExpand
 - hashtable.c, 63
 - hashtable.h, 29
- library/include/gui.h, 20
- library/include/hashtable.h, 24
- library/include/infoHashtable.h, 30
- library/include/infoList.h, 33
- library/include/list.h, 35
- library/include/message.h, 39
- library/include/queue.h, 42
- library/include/redblackTree.h, 44
- library/include/TBST.h, 46
- library/include/tcp_socket.h, 48
- library/include/TInfoBST.h, 52
- library/include/treeNode.h, 53
- library/src/gui.c, 55
- library/src/hashtable.c, 58
- library/src/infoList.c, 63
- library/src/list.c, 65
- library/src/message.c, 69
- library/src/queue.c, 71
- library/src/redblackTree.c, 73
- library/src/TBST.c, 74
- library/src/tcp_socket.c, 75
- library/src/TInfoBST.c, 79
- library/src/treeNode.c, 80
- library/test.c, 82
- list.c
 - listCreate, 66
 - listDelete, 66
 - listDestroy, 67
 - listInsert, 67
 - listPrint, 67
 - listSearch, 68
 - nodeCreate, 68
 - nodeDestroy, 68
- list.h
 - listCreate, 37
 - listDelete, 37
 - listDestroy, 38
 - listInsert, 38
 - listPrint, 38
 - listSearch, 39
 - THLNode, 37
- listCreate
 - list.c, 66
 - list.h, 37
- listDelete
 - list.c, 66
 - list.h, 37
- listDestroy
 - list.c, 67
 - list.h, 38
- listInsert
 - list.c, 67
 - list.h, 38
- listPrint
 - list.c, 67
 - list.h, 38
- listSearch
 - list.c, 68
 - list.h, 39
- MESSAGE
 - message.h, 41
- message.c
 - message_code_constructor, 70
 - message_constructor, 70
 - message_create, 71
 - message_destroy, 71
- message.h
 - MESSAGE, 41
 - message_code_constructor, 41
 - message_constructor, 41
 - message_create, 42
 - message_destroy, 42
 - MESSAGE_STRUCT, 41
- message_code_constructor
 - message.c, 70
 - message.h, 41
- message_constructor
 - message.c, 70
 - message.h, 41
- message_create
 - message.c, 71
 - message.h, 42
- message_destroy
 - message.c, 71
 - message.h, 42
- MESSAGE_STRUCT
 - message.h, 41
- nodeCreate
 - list.c, 68
- nodeDestroy
 - list.c, 68
- RBTree, 7
- SBSTnode, 8
- SBucket, 8
- server.c
 - server_manage_client, 85

- server_sendToAll, [86](#)
- server_sendUsersTo, [86](#)
- server.h
 - server_manage_client, [20](#)
 - server_sendToAll, [20](#)
 - server_sendUsersTo, [20](#)
 - ThreadArgs, [19](#)
- server_manage_client
 - server.c, [85](#)
 - server.h, [20](#)
- server_sendToAll
 - server.c, [86](#)
 - server.h, [20](#)
- server_sendUsersTo
 - server.c, [86](#)
 - server.h, [20](#)
- SHashTable, [9](#)
- SHLNode, [9](#)
- SMessage, [10](#)
- src/client.c, [83](#)
- src/server.c, [84](#)
- STCPSocket, [10](#)
- SThreadArgs, [11](#)
- TBucket
 - hashtable.h, [26](#)
- tcp_socket.c
 - tcp_socket_client_connect, [76](#)
 - tcp_socket_create, [77](#)
 - tcp_socket_destroy, [77](#)
 - tcp_socket_recv_message, [77](#)
 - tcp_socket_send_message, [78](#)
 - tcp_socket_server_accept, [78](#)
 - tcp_socket_server_listen, [79](#)
- tcp_socket.h
 - tcp_socket_client_connect, [50](#)
 - tcp_socket_create, [50](#)
 - tcp_socket_destroy, [50](#)
 - tcp_socket_recv_message, [51](#)
 - tcp_socket_send_message, [51](#)
 - tcp_socket_server_accept, [51](#)
 - tcp_socket_server_listen, [52](#)
- tcp_socket_client_connect
 - tcp_socket.c, [76](#)
 - tcp_socket.h, [50](#)
- tcp_socket_create
 - tcp_socket.c, [77](#)
 - tcp_socket.h, [50](#)
- tcp_socket_destroy
 - tcp_socket.c, [77](#)
 - tcp_socket.h, [50](#)
- tcp_socket_recv_message
 - tcp_socket.c, [77](#)
 - tcp_socket.h, [51](#)
- tcp_socket_send_message
 - tcp_socket.c, [78](#)
 - tcp_socket.h, [51](#)
- tcp_socket_server_accept
 - tcp_socket.c, [78](#)
- tcp_socket.h, [51](#)
- tcp_socket_server_listen
 - tcp_socket.c, [79](#)
 - tcp_socket.h, [52](#)
- THLNode
 - list.h, [37](#)
- ThreadArgs
 - server.h, [19](#)
- TInfoHashtable, [11](#)
- TInfoList
 - infoList.h, [34](#)
- TKeyHashtable
 - infoHashtable.h, [31](#)
- TQueue, [12](#)
- Treenode, [13](#)