Axiom Enterprises - Route Logistics

0.2

Generated by Doxygen 1.8.6

Wed Mar 26 2014 22:31:27

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

File Index

4.1 File List

Here is a list of all files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 basic_window Namespace Reference

Functions

- def main

5.1.1 Function Documentation

5.1.1.1 def basic_window.main ( )

5.2 hello Namespace Reference

Functions

- def hello
- def multiply

5.2.1 Function Documentation

5.2.1.1 def hello.hello ( )

5.2.1.2 def hello.multiply (*a*, *b*)

5.3 python_client_test Namespace Reference

Variables

- string HOST = '127.0.0.1'
- int PORT = 3490
- tuple s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
- tuple data = s.recv(4)

5.3.1 Variable Documentation

5.3.1.1 tuple python_client_test.data = s.recv(4)

5.3.1.2 string python_client_test.HOST = '127.0.0.1'

5.3.1.3 int python_client_test.PORT = 3490

5.3.1.4 tuple python_client_test.s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

5.4 python_server_test Namespace Reference

Variables

- string HOST = '127.0.0.1'
- int PORT = 50007
- tuple s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
- tuple data = conn.recv(1024)

5.4.1 Variable Documentation

5.4.1.1 tuple python_server_test.data = conn.recv(1024)

5.4.1.2 string python_server_test.HOST = '127.0.0.1'

5.4.1.3 int python_server_test.PORT = 50007

5.4.1.4 tuple python_server_test.s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

Chapter 6

Class Documentation

6.1 account Struct Reference

`#include "python_ui.h"`

Public Attributes

- char * username
- char * password

6.1.1 Member Data Documentation

6.1.1.1 char* account::password

6.1.1.2 char* account::username

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

- tools/Sockets/python_ui.h

6.2 Connection< Type > Class Template Reference

`#include "Graph.h"`

Public Member Functions

- Connection ()

6.2.1 Constructor & Destructor Documentation

6.2.1.1 template<class Type > Connection< Type >::Connection ( )

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

- tools/Graph.h

6.3 LinkedList< Type > Class Template Reference

`#include "LinkedList.h"`

Public Member Functions

- LinkedList ()
- ~LinkedList ()
- void appendNode (double latitude, double longitude)
- void appendNode (Node< Type > *n)
- Type * getNextNode ()
- Type * getNodeAt (int NodeIndex)
- int getListLength ()
- void loopThroughFrom (int NodeIndex)
- void fromStart ()
- void displayNodes ()
- void displayNodesR ()
- void emptyList ()

Private Attributes

- Node< Type > * front
- Node< Type > * back
- Node< Type > * currentNode
- int list_length

6.3.1 Constructor & Destructor Documentation

6.3.1.1 template<class Type > LinkedList< Type >::LinkedList ( )

6.3.1.2 template<class Type > LinkedList< Type >::~LinkedList ( )

6.3.2 Member Function Documentation

6.3.2.1 template<class Type > void LinkedList< Type >::appendNode (double *latitude*, double *longitude*)

6.3.2.2 template<class Type> void LinkedList< Type >::appendNode (Node< Type > * *n*)

6.3.2.3 template<class Type > void LinkedList< Type >::displayNodes ( )

6.3.2.4 template<class Type > void LinkedList< Type >::displayNodesR ( )

6.3.2.5 template<class Type > void LinkedList< Type >::emptyList ( )

6.3.2.6 template<class Type> void LinkedList< Type >::fromStart ( ) [inline]

6.3.2.7 template<class Type> int LinkedList< Type >::getListLength ( ) [inline]

6.3.2.8 template<class Type > Type * LinkedList< Type >::getNextNode ( )

6.3.2.9 template<class Type > Type * LinkedList< Type >::getNodeAt (int *NodeIndex*)

6.3.2.10 template<class Type > void LinkedList< Type >::loopThroughFrom (int *NodeIndex*)

6.3.3 Member Data Documentation

6.3.3.1 template<class Type> Node<Type>* LinkedList< Type >::back [private]

6.3.3.2 template<class Type> Node<Type>* LinkedList< Type >::currentNode [private]

6.3.3.3 template<class Type> Node<Type>* LinkedList< Type >::front [private]

6.3.3.4 template<class Type> int LinkedList< Type >::list_length [private]

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

- tools/LinkedList.h

6.4 location Struct Reference

`#include "server_utils.h"`

Public Attributes

- double id
- double latitude
- double longitude

6.4.1 Member Data Documentation

6.4.1.1 double location::id

6.4.1.2 double location::latitude

6.4.1.3 double location::longitude

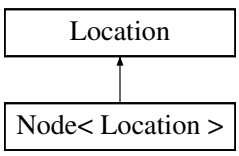
The documentation for this struct was generated from the following file:

- tools/Sockets/server_utils.h

6.5 Location Class Reference

`#include "Location.h"`

Inheritance diagram for Location:



Public Member Functions

- Location (double latitude=0, double longitude=0)

- ~Location ()
- void setStart ()
- bool startLocation ()
- void notStart ()
- double getLong ()
- double getLat ()
- void printCoords ()
- double getDistanceTo (Location &loc)

Public Attributes

- bool isStart
- int locationID
- double y
- double x

6.5.1 Constructor & Destructor Documentation

6.5.1.1 Location::Location (double *latitude* = 0, double *longitude* = 0)

6.5.1.2 Location::~Location ( )

6.5.2 Member Function Documentation

6.5.2.1 double Location::getDistanceTo (Location & *loc*)

6.5.2.2 double Location::getLat ( ) [inline]

6.5.2.3 double Location::getLong ( ) [inline]

6.5.2.4 void Location::notStart ( ) [inline]

6.5.2.5 void Location::printCoords ( ) [inline]

6.5.2.6 void Location::setStart ( ) [inline]

6.5.2.7 bool Location::startLocation ( ) [inline]

6.5.3 Member Data Documentation

6.5.3.1 bool Location::isStart

6.5.3.2 int Location::locationID

6.5.3.3 double Location::x

6.5.3.4 double Location::y

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

- Backend/Salesman/Location.h
- Backend/Salesman/Location.cpp

6.6 Matrix Class Reference

A class that defines a matrix.

`#include "Matrices.h"`

Public Member Functions

- Matrix (int elements)
- ~Matrix ()
- bool setElement (int row, int column, double value)
- double getElement (int row, int column)
- void showMatrix ()
- int numberOfColumns ()

Private Attributes

- double * values
- int number_of_columns
- int number_of_elements

6.6.1 Detailed Description

A class that defines a matrix.

This class defines a matrix of arbitrary dimesions, I think the current implementation restricts this to a square matrix (nxn) where n is the square root of the value given to the constructor.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 Matrix::Matrix (int *elements*)

6.6.2.2 Matrix::~Matrix ( )

6.6.3 Member Function Documentation

6.6.3.1 double Matrix::getElement (int *row,* int *column*)

6.6.3.2 int Matrix::numberOfColumns ( ) [inline]

6.6.3.3 bool Matrix::setElement (int *row,* int *column,* double *value*)

6.6.3.4 void Matrix::showMatrix ( )

6.6.4 Member Data Documentation

6.6.4.1 int Matrix::number_of_columns [private]

6.6.4.2 int Matrix::number_of_elements [private]

6.6.4.3 double* Matrix::values [private]

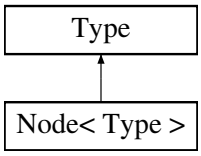
The documentation for this class was generated from the following files:

- tools/Matrices.h
- tools/Matrices.cpp

6.7 Node< Type > Class Template Reference

`#include "LinkedList.h"`

Inheritance diagram for Node< Type >:



Public Member Functions

- Node (double latitude, double longitude)

Public Attributes

- Node< Type > * next
- Node< Type > * previous

6.7.1 Constructor & Destructor Documentation

6.7.1.1 template<class Type> Node< Type >::Node (double *latitude,* double *longitude*) [inline]

6.7.2 Member Data Documentation

6.7.2.1 template<class Type> Node<Type>* Node< Type >::next

6.7.2.2 template<class Type> Node<Type>* Node< Type >::previous

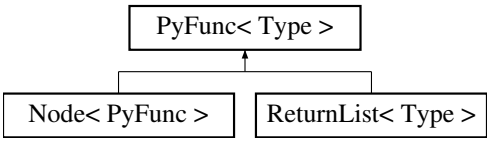
The documentation for this class was generated from the following file:

- tools/LinkedList.h

6.8 PyFunc< Type > Class Template Reference

`#include "PyFunc.h"`

Inheritance diagram for PyFunc< Type >:



Public Member Functions

- PyFunc (const char *Module, const char *FuncName, char *FileName)
- ~PyFunc ()
- bool validFunc ()

Protected Attributes

- PyObject * pModule
- PyObject * pName
- PyObject * pFunc
- PyObject * pList
- PyObject * pValue
- PyObject * pListLength
- PyObject pListItem
- bool isValid

6.8.1 Constructor & Destructor Documentation

6.8.1.1 template<class Type > PyFunc< Type >::PyFunc (const char * *Module*, const char * *FuncName*, char * *FileName*)

6.8.1.2 template<class Type > PyFunc< Type >::~PyFunc ( )

6.8.2 Member Function Documentation

6.8.2.1 template<class Type > bool PyFunc< Type >::validFunc ( ) [inline]

6.8.3 Member Data Documentation

6.8.3.1 template<class Type > bool PyFunc< Type >::isValid [protected]

6.8.3.2 template<class Type > PyObject * PyFunc< Type >::pFunc [protected]

6.8.3.3 template<class Type > PyObject * PyFunc< Type >::pList [protected]

6.8.3.4 template<class Type > PyObject PyFunc< Type >::pListItem [protected]

6.8.3.5 template<class Type > PyObject* PyFunc< Type >::pListLength [protected]

6.8.3.6 template<class Type > PyObject* PyFunc< Type >::pModule [protected]

6.8.3.7 template<class Type > PyObject * PyFunc< Type >::pName [protected]

6.8.3.8 template<class Type > PyObject * PyFunc< Type >::pValue [protected]

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

- tools/Python/PyFunc.h
- tools/Python/.PyFunc.cpp

6.9 PyI Class Reference

```
#include ".PyI.h"
```

Public Member Functions

- PyI ()
- ~PyI ()
- void addFunction (PyFunc *pFunc)

Private Attributes

- LinkedList< PyFunc > * funcList

6.9.1 Constructor & Destructor Documentation

6.9.1.1 PyI::PyI ( )

6.9.1.2 PyI::~PyI ( )

6.9.2 Member Function Documentation

6.9.2.1 void PyI::addFunction (PyFunc * *pFunc*)

6.9.3 Member Data Documentation

6.9.3.1 LinkedList<PyFunc>* PyI::funcList [private]

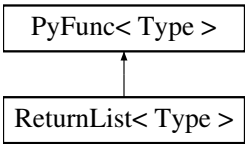
The documentation for this class was generated from the following files:

- tools/Python/.PyI.h
- tools/Python/.PyI.cpp

6.10 ReturnList< Type > Class Template Reference

`#include "Functions.h"`

Inheritance diagram for ReturnList< Type >:



Public Member Functions

- ReturnList (const char *ModuleName, const char *FuncName, char *FileName)
- Type * callFunction ()

Additional Inherited Members

6.10.1 Constructor & Destructor Documentation

6.10.1.1 template<class Type > ReturnList< Type >::ReturnList (const char * *ModuleName,* const char * *FuncName,* char * *FileName*) [inline]

6.10.2 Member Function Documentation

6.10.2.1 template<class Type > Type * ReturnList< Type >::callFunction ( )

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

- tools/Python/Functions.h
- tools/Python/.Functions.cpp

6.11 Salesman Class Reference

```
#include "Salesman.h"
```

Public Member Functions

- Salesman ()
- ~Salesman ()
- void addLocation (double longitude, double latitude)
- void showLocations ()
- void showRoute ()
- bool populateMatrix ()
- void calculateRoute ()

Private Attributes

- LinkedList< Location > * locations
- LinkedList< Location > * route
- Matrix * distanceMatrix
- bool hasMatrix

6.11.1 Constructor & Destructor Documentation

6.11.1.1 Salesman::Salesman ( )

6.11.1.2 Salesman::~Salesman ( )

6.11.2 Member Function Documentation

6.11.2.1 void Salesman::addLocation (double *longitude,* double *latitude*)

6.11.2.2 void Salesman::calculateRoute ( )

6.11.2.3 bool Salesman::populateMatrix ( )

6.11.2.4 void Salesman::showLocations ( )

6.11.2.5 void Salesman::showRoute ( )

6.11.3 Member Data Documentation

6.11.3.1 Matrix* Salesman::distanceMatrix [private]

6.11.3.2 bool Salesman::hasMatrix [private]

6.11.3.3 LinkedList<Location>* Salesman::locations [private]

6.11.3.4 LinkedList<Location>* Salesman::route [private]

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

- Backend/Salesman/Salesman.h
- Backend/Salesman/Salesman.cpp

Chapter 7

File Documentation

7.1 Backend/Salesman/Location.cpp File Reference

```
#include "Location.h"
```

7.2 Backend/Salesman/Location.h File Reference

```
#include <iostream>
#include <cmath>
#include "../../tools/LinkedList.h"
```

Classes

- class Location

7.3 Backend/Salesman/Salesman.cpp File Reference

```
#include "Salesman.h"
```

7.4 Backend/Salesman/Salesman.h File Reference

```
#include "Location.h"
#include "../../tools/LinkedList.h"
#include "../../tools/Matrices.h"
```

Classes

- class Salesman

7.5 Backend/Salesman/test.cpp File Reference

```
#include "Location.h"
#include "Salesman.h"
```

Functions

- int main ()

7.5.1 Function Documentation

7.5.1.1 int main ( )

7.6 tests/test.cpp File Reference

```
#include <python2.7/Python.h>
#include <iostream>
```

Functions

- int main ()

7.6.1 Function Documentation

7.6.1.1 int main ( )

7.7 tools/Sockets/test.cpp File Reference

```
#include "server_utils.h"
```

Functions

- int main ()

7.7.1 Function Documentation

7.7.1.1 int main ( )

7.8 tools/test.cpp File Reference

```
#include "LinkedList.h"
```

Functions

- int main ()

7.8.1 Function Documentation

7.8.1.1 int main ( )

7.9 Backend/Salesman/test2.cpp File Reference

```
#include "Location.h"
```

Functions

- int main ()

7.9.1 Function Documentation

7.9.1.1 int main ( )

7.10 main.cpp File Reference

```
#include <iostream>
#include <thread>
#include "tools/Sockets/server.h"
```

Functions

- int main ()

7.10.1 Function Documentation

7.10.1.1 int main ( )

7.11 tests/basic_window.py File Reference

Namespaces

- basic_window

Functions

- def basic_window.main

7.12 tests/example.cpp File Reference

```
#include <python2.7/Python.h>
#include <iostream>
```

Functions

- int main (int argc, char ∗argv[])

7.12.1 Function Documentation

7.12.1.1 int main (int *argc,* char ∗ *argv[]*)

7.13 tests/hello.py File Reference

Namespaces

- hello

Functions

- def hello.hello
- def hello.multiply

7.14 tools/Graph.h File Reference

```
#include <iostream>
```

Classes

- class Connection< Type >

Functions

- template<class Type >
  class Connection< Type > Node ()
- Connection ()
- ∼Node ()

7.14.1 Function Documentation

7.14.1.1 Node::Connection ( )

7.14.1.2 template<class Type > class Connection< Type > Node ( )

7.14.1.3 ∼Node ( )

7.15 tools/LinkedList.h File Reference

```
#include <iostream>
#include <stdexcept>
```

Classes

- class Node< Type >
- class LinkedList< Type >

Variables

- Node Node

7.15.1 Variable Documentation

7.15.1.1 Node Node

7.16 tools/Matrices.cpp File Reference

```
#include "Matrices.h"
```

7.17 tools/Matrices.h File Reference

```
#include <iostream>
#include <cmath>
```

Classes

- class Matrix

  A class that defines a matrix.

7.18 tools/Python/.Functions.cpp File Reference

```
#include "Functions.h"
```

7.19 tools/Python/.PyFunc.cpp File Reference

```
#include "PyFunc.h"
```

7.20 tools/Python/.PyI.cpp File Reference

```
#include "PyI.h"
#include "PyFunc.h"
```

7.21 tools/Python/.PyI.h File Reference

```
#include <python2.7/Python.h>
#include "PyFunc.h"
#include "../tools/LinkedList.h"
```

Classes

- class PyI

7.22 tools/Python/Functions.h File Reference

```
#include "PyFunc.h"
```

Classes

- class ReturnList< Type >

7.23 tools/Python/PyFunc.h File Reference

```
#include <python2.7/Python.h>
#include <iostream>
```

Classes

- class PyFunc< Type >

7.24 tools/Sockets/client.c File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <netdb.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <sys/socket.h>
#include <arpa/inet.h>
```

Macros

- #define PORT "3490"
- #define MAXDATASIZE 100

Functions

- void ∗ get_in_addr (struct sockaddr ∗sa)
- int main (int argc, char ∗argv[])

7.24.1 Macro Definition Documentation

7.24.1.1 #define MAXDATASIZE 100

7.24.1.2 #define PORT "3490"

7.24.2 Function Documentation

7.24.2.1 void∗ get_in_addr (struct sockaddr ∗ *sa*)

7.24.2.2 int main (int *argc,* char ∗ *argv[]*)

7.25 tools/Sockets/python_client_test.py File Reference

Namespaces

- python_client_test

Variables

- string python_client_test.HOST = '127.0.0.1'
- int python_client_test.PORT = 3490
- tuple python_client_test.s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
- tuple python_client_test.data = s.recv(4)

7.26 tools/Sockets/python_server_test.py File Reference

Namespaces

- python_server_test

Variables

- string python_server_test.HOST = '127.0.0.1'
- int python_server_test.PORT = 50007
- tuple python_server_test.s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
- tuple python_server_test.data = conn.recv(1024)

7.27 tools/Sockets/python_ui.c File Reference

```
#include "python_ui.h"
```

Functions

- int handleUI (int their_socket)
- int getNumLines (FILE *file)
- int loadAccounts (FILE *file, struct account *accounts, int num)
- int loginSuccessful ()

7.27.1 Function Documentation

7.27.1.1 int getNumLines (FILE * *file*)

7.27.1.2 int handleUI (int *their_socket*)

7.27.1.3 int loadAccounts (FILE * *file*, struct **account** * *accounts*, int *num*)

7.27.1.4 int loginSuccessful ( )

7.28 tools/Sockets/python_ui.h File Reference

```
#include <string.h>
#include "server_utils.h"
```

Classes

- struct account

Functions

- int handleUI (int their_socket)
- int loginSuccesful ()
- int getNumLines (FILE *file)
- int loadAccounts (FILE *file, struct account *accounts, int num)

7.28.1 Function Documentation

7.28.1.1 int getNumLines (FILE * *file*)

7.28.1.2 int handleUI (int *their_socket*)

7.28.1.3 int loadAccounts (FILE * *file*, struct **account** * *accounts*, int *num*)

7.28.1.4 int loginSuccesful ( )

7.29 tools/Sockets/server.c File Reference

```
#include "server_utils.h"
```

Functions

- int run_server (int sock)

7.29.1 Function Documentation

7.29.1.1 int run_server (int *sock*)

7.30 tools/Sockets/server.h File Reference

```
#include "server_utils.h"
#include "python_ui.h"
```

Functions

- int run_server (int sock)

7.30.1 Function Documentation

7.30.1.1 int run_server (int *sock*)

7.31 tools/Sockets/server_utils.c File Reference

```
#include "server_utils.h"
```

Functions

- void sigchld_handler (int s)
- void * get_in_addr (struct sockaddr *sa)
- int new_socket (int portNum)
- int receiveData (int their_socket, char *data, int data_size)

7.31.1 Function Documentation

7.31.1.1 void* get_in_addr (struct sockaddr * *sa*)

7.31.1.2 int new_socket (int *portNum*)

7.31.1.3 int receiveData (int *their_socket,* char * *data,* int *data_size*)

7.31.1.4 void sigchld_handler (int *s*)

7.32 tools/Sockets/server_utils.h File Reference

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <errno.h>
#include <string.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <arpa/inet.h>
#include <sys/wait.h>
#include <signal.h>
```

Classes

- struct location

Macros

- #define PORT "3490"
- #define BACKLOG 10
- #define UI 1
- #define APP 2

Functions

- int receiveData (int their_socket, char ∗data, int data_size)
- void sigchld_handler (int s)
- int new_socket (int portNum)
- void ∗ get_in_addr (struct sockaddr ∗sa)

7.32.1 Macro Definition Documentation

7.32.1.1 #define APP 2

7.32.1.2 #define BACKLOG 10

7.32.1.3 #define PORT "3490"

7.32.1.4 #define UI 1

7.32.2 Function Documentation

7.32.2.1 void∗ get_in_addr (struct sockaddr ∗ *sa*)

7.32.2.2 int new_socket (int *portNum*)

7.32.2.3 int receiveData (int *their_socket,* char ∗ *data,* int *data_size*)

7.32.2.4 void sigchld_handler (int *s*)

Index