

labo_11_schaufelberger_yannick_gallay_david

Generated by Doxygen 1.8.13

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	map.h File Reference	3
2.1.1	Typedef Documentation	4
2.1.1.1	Axe	4
2.1.1.2	Map	4
2.1.2	Enumeration Type Documentation	4
2.1.2.1	MapState	4
2.1.3	Function Documentation	5
2.1.3.1	addLake()	5
2.1.3.2	addRandomLake() [1/2]	5
2.1.3.3	addRandomLake() [2/2]	5
2.1.3.4	addRandomStart() [1/2]	5
2.1.3.5	addRandomStart() [2/2]	6
2.1.3.6	addRandomTreasure()	6
2.1.3.7	addStart()	6
2.1.3.8	addTreasure()	6
2.1.3.9	displayWorld()	6
2.1.3.10	getEmptyMap()	6
2.1.3.11	getHeight()	7
2.1.3.12	getMapValue()	7
2.1.3.13	getWidth()	7

2.1.3.14	<code>initWorld()</code>	7
2.1.3.15	<code>setMapValue()</code>	7
2.1.4	Variable Documentation	7
2.1.4.1	<code>NUMBER_OF_LAKE</code>	7
2.2	<code>searchers.h</code> File Reference	8
2.2.1	Typedef Documentation	9
2.2.1.1	<code>Searcher</code>	9
2.2.1.2	<code>SearcherList</code>	9
2.2.2	Enumeration Type Documentation	9
2.2.2.1	<code>ResearcherStatus</code>	9
2.2.3	Function Documentation	10
2.2.3.1	<code>displaySearcherList()</code>	10
2.2.3.2	<code>getStatus()</code>	10
2.2.3.3	<code>getStatusString()</code>	10
2.2.3.4	<code>getSteps()</code>	11
2.2.3.5	<code>initSearcher()</code>	11
2.2.3.6	<code>setStatus()</code>	11
2.2.3.7	<code>setSteps()</code>	12
2.3	<code>treasure.h</code> File Reference	12
2.3.1	Function Documentation	13
2.3.1.1	<code>getStatistics()</code>	13
2.3.1.2	<code>runSearcher()</code>	14
2.3.1.3	<code>runSimulation()</code>	14
2.4	<code>utilities.h</code> File Reference	14
2.4.1	Function Documentation	15
2.4.1.1	<code>askForNumberOfSimulation()</code>	15
2.4.1.2	<code>askForRestart()</code>	15
2.4.1.3	<code>getRandomInRange()</code>	15
2.4.2	Variable Documentation	15
2.4.2.1	<code>RESTART_CHAR</code>	15
2.4.2.2	<code>STOP_CHAR</code>	15

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

map.h	3
searchers.h	8
treasure.h	12
utilities.h	14

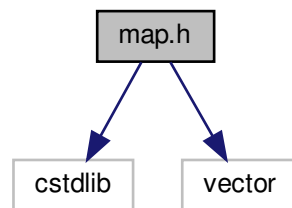
Chapter 2

File Documentation

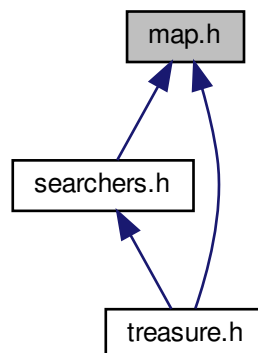
2.1 map.h File Reference

```
#include <cstdlib>
#include <vector>
```

Include dependency graph for map.h:



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



Typedefs

- typedef std::vector< [MapState](#) > [Axe](#)
- typedef std::vector< [Axe](#) > [Map](#)

Enumerations

- enum [MapState](#) {
[MS_OUT](#), [MS_EARTH](#), [MS_WATER](#), [MS_TREASURE](#),
[MS_START](#) }

Functions

- size_t [getHeight](#) (const [Map](#) &map)
- size_t [getWidth](#) (const [Map](#) &map)
- [Map](#) [getEmptyMap](#) (size_t height, size_t width)
- [MapState](#) [getMapValue](#) (const [Map](#) &map, size_t x, size_t y)
- bool [setMapValue](#) ([Map](#) &map, size_t x, size_t y, [MapState](#) value)
- bool [addTreasure](#) ([Map](#) &map, size_t height, size_t width)
- void [addRandomTreasure](#) ([Map](#) &map)
- bool [addLake](#) ([Map](#) &map, size_t originX, size_t originY, size_t radius)
- void [addRandomLake](#) ([Map](#) &map)
- void [addRandomLake](#) ([Map](#) &map, size_t maxRadius)
- bool [addStart](#) ([Map](#) &map, size_t x, size_t y)
- void [addRandomStart](#) ([Map](#) &map)
- void [addRandomStart](#) ([Map](#) &map, size_t &x, size_t &y)
- [Map](#) [initWorld](#) (size_t height, size_t width, size_t &x, size_t &y)
- void [displayWorld](#) (const [Map](#) &map)

Variables

- const int [NUMBER_OF_LAKE](#) = 3

2.1.1 Typedef Documentation

2.1.1.1 [Axe](#)

```
typedef std::vector<MapState> Axe
```

2.1.1.2 [Map](#)

```
typedef std::vector<Axe> Map
```

2.1.2 Enumeration Type Documentation

2.1.2.1 [MapState](#)

```
enum MapState
```


Enumerator

MS_OUT	
MS_EARTH	
MS_WATER	
MS_TREASURE	
MS_START	

2.1.3 Function Documentation

2.1.3.1 addLake()

```
bool addLake (
    Map & map,
    size_t originX,
    size_t originY,
    size_t radius )
```

2.1.3.2 addRandomLake() [1/2]

```
void addRandomLake (
    Map & map )
```

2.1.3.3 addRandomLake() [2/2]

```
void addRandomLake (
    Map & map,
    size_t maxRadius )
```

2.1.3.4 addRandomStart() [1/2]

```
void addRandomStart (
    Map & map )
```

2.1.3.5 addRandomStart() [2/2]

```
void addRandomStart (
    Map & map,
    size_t & x,
    size_t & y )
```

2.1.3.6 addRandomTreasure()

```
void addRandomTreasure (
    Map & map )
```

2.1.3.7 addStart()

```
bool addStart (
    Map & map,
    size_t x,
    size_t y )
```

2.1.3.8 addTreasure()

```
bool addTreasure (
    Map & map,
    size_t height,
    size_t width )
```

2.1.3.9 displayWorld()

```
void displayWorld (
    const Map & map )
```

2.1.3.10 getEmptyMap()

```
Map getEmptyMap (
    size_t height,
    size_t width )
```

2.1.3.11 getHeight()

```
size_t getHeight (
    const Map & map )
```

2.1.3.12 getMapValue()

```
MapState getMapValue (
    const Map & map,
    size_t x,
    size_t y )
```

2.1.3.13 getWidth()

```
size_t getWidth (
    const Map & map )
```

2.1.3.14 initWorld()

```
Map initWorld (
    size_t height,
    size_t width,
    size_t & x,
    size_t & y )
```

2.1.3.15 setMapValue()

```
bool setMapValue (
    Map & map,
    size_t x,
    size_t y,
    MapState value )
```

2.1.4 Variable Documentation

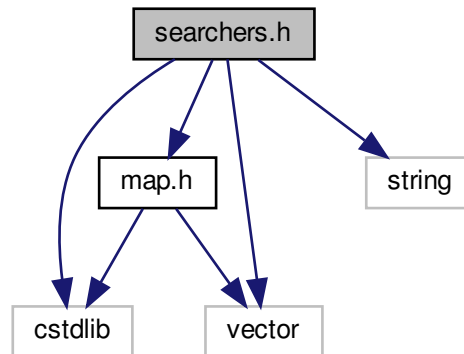
2.1.4.1 NUMBER_OF_LAKE

```
const int NUMBER_OF_LAKE = 3
```

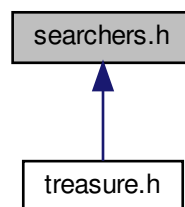
2.2 searchers.h File Reference

```
#include "map.h"  
#include <cstdlib>  
#include <vector>  
#include <string>
```

Include dependency graph for searchers.h:



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



Typedefs

- typedef std::vector< int > [Searcher](#)
- typedef std::vector< [Searcher](#) > [SearcherList](#)

Enumerations

- enum [ResearcherStatus](#) {
 [UNDEFINED](#), [RICH](#), [LOST](#), [DROWNED](#),
 [EXHAUSTED](#) }

Functions

- [Searcher](#) `initSearcher` (int stepsValue=0, [ResearcherStatus](#) statusValue=UNDEFINED)
initialize a searcher with given values
- int `getSteps` (const [Searcher](#) &searcher)
- int `getStatus` (const [Searcher](#) &searcher)
- std::string `getStatusString` (const [Searcher](#) &searcher)
- bool `setSteps` ([Searcher](#) &searcher, int value)
sets the steps of a searcher to the given value
- bool `setStatus` ([Searcher](#) &searcher, [ResearcherStatus](#) value)
sets the status of a searcher to the given value
- bool `displaySearcherList` ([SearcherList](#) &list)
displays the information of every searcher in the list

2.2.1 Typedef Documentation

2.2.1.1 Searcher

```
typedef std::vector<int> Searcher
```

2.2.1.2 SearcherList

```
typedef std::vector<Searcher> SearcherList
```

2.2.2 Enumeration Type Documentation

2.2.2.1 ResearcherStatus

```
enum ResearcherStatus
```

Enumerator

UNDEFINED	
RICH	
LOST	
DROWNED	
EXHAUSTED	

2.2.3 Function Documentation

2.2.3.1 displaySearcherList()

```
bool displaySearcherList (
    SearcherList & list )
```

displays the information of every searcher in the list

Parameters

<i>list</i>	
-------------	--

Returns

true if success

2.2.3.2 getStatus()

```
int getStatus (
    const Searcher & searcher )
```

Parameters

<i>searcher</i>	
-----------------	--

Returns

the status of a searcher

2.2.3.3 getStatusString()

```
std::string getStatusString (
    const Searcher & searcher )
```

Parameters

<i>searcher</i>	
-----------------	--

Returns

a string containing the status of a searcher

2.2.3.4 getSteps()

```
int getSteps (
    const Searcher & searcher )
```

Parameters

<i>searcher</i>	
-----------------	--

Returns

the amount of steps taken by a searcher

2.2.3.5 initSearcher()

```
Searcher initSearcher (
    int stepsValue = 0,
    ResearcherStatus statusValue = UNDEFINED )
```

initialize a searcher with given values

Parameters

<i>stepsValue</i>	
<i>statusValue</i>	

Returns**2.2.3.6 setStatus()**

```
bool setStatus (
    Searcher & searcher,
    ResearcherStatus value )
```

sets the status of a searcher to the given value

Parameters

<i>searcher</i>	
<i>value</i>	

Returns

true if success

2.2.3.7 setSteps()

```
bool setSteps (
    Searcher & searcher,
    int value )
```

sets the steps of a searcher to the given value

Parameters

<i>searcher</i>	
<i>value</i>	

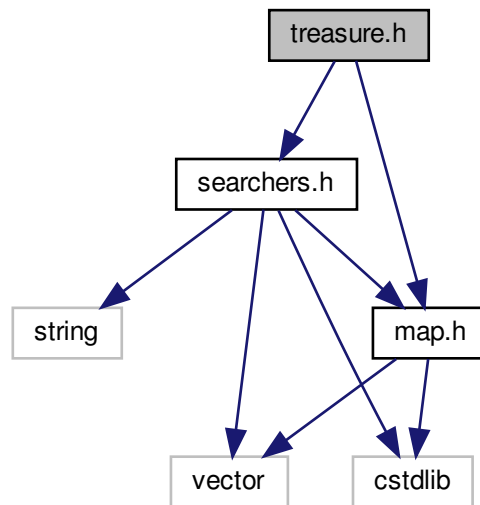
Returns

true if success

2.3 treasure.h File Reference

```
#include "map.h"
#include "searchers.h"
```


Include dependency graph for treasure.h:



Functions

- [SearcherList runSimulation](#) (const [Map](#) &map, size_t startX, size_t startY, int nbSimulation)
Simulates nbSimulation searchers starting at the coordinates startX and startY on the map.
- void [runSearcher](#) (const [Map](#) &map, size_t startX, size_t startY, [Searcher](#) &searcher)
walks a searcher on the map, defines the status and steps taken at the end.
- bool [getStatistics](#) (const [SearcherList](#) &list, double &probability, double &avgSteps)
calculates the probability for a searcher to find the treasure, and the average steps taken to find it

2.3.1 Function Documentation

2.3.1.1 getStatistics()

```

bool getStatistics (
    const SearcherList & list,
    double & probability,
    double & avgSteps )

```

calculates the probability for a searcher to find the treasure, and the average steps taken to find it

Parameters

<i>list</i>	
<i>probability</i>	
<i>avgSteps</i>	

Returns

2.3.1.2 runSearcher()

```
void runSearcher (
    const Map & map,
    size_t startX,
    size_t startY,
    Searcher & searcher )
```

walks a searcher on the map, defines the status and steps taken at the end.

Parameters

<i>map</i>	
<i>startX</i>	
<i>startY</i>	
<i>searcher</i>	

2.3.1.3 runSimulation()

```
SearcherList runSimulation (
    const Map & map,
    size_t startX,
    size_t startY,
    int nbSimulation )
```

Simulates nbSimulation searchers starting at the coordinates startX and startY on the map.

Parameters

<i>map</i>	
<i>startX</i>	
<i>startY</i>	
<i>nbSimulation</i>	

Returns

a list containing the status and steps of every simulated searcher

2.4 utilities.h File Reference

Functions

- int [getRandomInRange](#) (int max, int min=0)

- bool `askForRestart()`
This function keeps asking as long as the user enters anything else than `RESTART_CHAR` or `STOP_CHAR`.
- int `askForNumberOfSimulation()`

Variables

- const char `RESTART_CHAR` = 'Y'
- const char `STOP_CHAR` = 'N'

2.4.1 Function Documentation

2.4.1.1 `askForNumberOfSimulation()`

```
int askForNumberOfSimulation ( )
```

2.4.1.2 `askForRestart()`

```
bool askForRestart ( )
```

This function keeps asking as long as the user enters anything else than `RESTART_CHAR` or `STOP_CHAR`.

Returns

true if the user has entered `RESTART_CHAR` and false if the user has entered `STOP_CHAR`

2.4.1.3 `getRandomInRange()`

```
int getRandomInRange (
    int max,
    int min = 0 )
```

2.4.2 Variable Documentation

2.4.2.1 `RESTART_CHAR`

```
const char RESTART_CHAR = 'Y'
```

2.4.2.2 `STOP_CHAR`

```
const char STOP_CHAR = 'N'
```


Index

- addLake
 - map.h, 5
- addRandomLake
 - map.h, 5
- addRandomStart
 - map.h, 5
- addRandomTreasure
 - map.h, 6
- addStart
 - map.h, 6
- addTreasure
 - map.h, 6
- askForNumberOfSimulation
 - utilities.h, 15
- askForRestart
 - utilities.h, 15
- Axe
 - map.h, 4
- displaySearcherList
 - searchers.h, 10
- displayWorld
 - map.h, 6
- getEmptyMap
 - map.h, 6
- getHeight
 - map.h, 6
- getMapValue
 - map.h, 7
- getRandomInRange
 - utilities.h, 15
- getStatistics
 - treasure.h, 13
- getStatus
 - searchers.h, 10
- getStatusString
 - searchers.h, 10
- getSteps
 - searchers.h, 11
- getWidth
 - map.h, 7
- initSearcher
 - searchers.h, 11
- initWorld
 - map.h, 7
- Map
 - map.h, 4

- map.h, 3
 - addLake, 5
 - addRandomLake, 5
 - addRandomStart, 5
 - addRandomTreasure, 6
 - addStart, 6
 - addTreasure, 6
 - Axe, 4
 - displayWorld, 6
 - getEmptyMap, 6
 - getHeight, 6
 - getMapValue, 7
 - getWidth, 7
 - initWorld, 7
 - Map, 4
 - MapState, 4
 - NUMBER_OF_LAKE, 7
 - setMapValue, 7
- MapState
 - map.h, 4
- NUMBER_OF_LAKE
 - map.h, 7
- RESTART_CHAR
 - utilities.h, 15
- ResearcherStatus
 - searchers.h, 9
- runSearcher
 - treasure.h, 14
- runSimulation
 - treasure.h, 14
- STOP_CHAR
 - utilities.h, 15
- Searcher
 - searchers.h, 9
- SearcherList
 - searchers.h, 9
- searchers.h, 8
 - displaySearcherList, 10
 - getStatus, 10
 - getStatusString, 10
 - getSteps, 11
 - initSearcher, 11
 - ResearcherStatus, 9
 - Searcher, 9
 - SearcherList, 9
 - setStatus, 11
 - setSteps, 12

- setMapValue
 - map.h, [7](#)
- setStatus
 - searchers.h, [11](#)
- setSteps
 - searchers.h, [12](#)
- treasure.h, [12](#)
 - getStatistics, [13](#)
 - runSearcher, [14](#)
 - runSimulation, [14](#)
- utilities.h, [14](#)
 - askForNumberOfSimulation, [15](#)
 - askForRestart, [15](#)
 - getRandomInRange, [15](#)
 - RESTART_CHAR, [15](#)
 - STOP_CHAR, [15](#)