

AlgorithmAnalysis

1.0

Generated by Doxygen 1.7.5.1

Fri Dec 14 2012 21:37:28

Contents

1	File Index	1
1.1	File List	1
2	File Documentation	3
2.1	src/aux.cpp File Reference	3
2.1.1	Function Documentation	3
2.1.1.1	checkArrayEquality	3
2.1.1.2	checkIsOrdered	3
2.1.1.3	checkSameLength	3
2.1.1.4	copyArray	3
2.1.1.5	createRandomArray	3
2.1.1.6	printArrayContents	4
2.2	src/aux.h File Reference	4
2.2.1	Function Documentation	4
2.2.1.1	checkArrayEquality	4
2.2.1.2	checkIsOrdered	4
2.2.1.3	copyArray	4
2.2.1.4	createRandomArray	4
2.2.1.5	printArrayContents	4
2.3	src/generators.cpp File Reference	4
2.3.1	Function Documentation	5
2.3.1.1	generateAllFiles	5
2.3.1.2	generateBubbleSortDataFile	5
2.3.1.3	generateBubbleSortTime	5
2.3.1.4	generateInsertionSortDataFile	5
2.3.1.5	generateInsertionSortTime	5

2.3.1.6	generateQuickSortDataFile	5
2.3.1.7	generateQuickSortTime	5
2.3.1.8	generateSelectionSortDataFile	5
2.3.1.9	generateSelectionSortTime	5
2.4	src/generators.h File Reference	5
2.4.1	Function Documentation	5
2.4.1.1	generateAllFiles	5
2.4.1.2	generateBubbleSortDataFile	6
2.4.1.3	generateBubbleSortTime	6
2.4.1.4	generateInsertionSortDataFile	6
2.4.1.5	generateInsertionSortTime	6
2.4.1.6	generateQuickSortDataFile	6
2.4.1.7	generateQuickSortTime	6
2.4.1.8	generateSelectionSortDataFile	6
2.4.1.9	generateSelectionSortTime	6
2.5	src/interaction.cpp File Reference	6
2.5.1	Function Documentation	6
2.5.1.1	chooseAlgorithm	6
2.5.1.2	getGap	6
2.5.1.3	getMaximumInteger	6
2.5.1.4	getProblemSize	6
2.6	src/interaction.h File Reference	7
2.6.1	Function Documentation	7
2.6.1.1	chooseAlgorithm	7
2.6.1.2	getGap	7
2.6.1.3	getMaximumInteger	7
2.6.1.4	getProblemSize	7
2.7	src/main.cpp File Reference	7
2.7.1	Function Documentation	7
2.7.1.1	main	7
2.8	src/ordenacion.cpp File Reference	7
2.8.1	Function Documentation	8
2.8.1.1	bubbleSort	8
2.8.1.2	insertionSort	8

2.8.1.3	quickSort	8
2.8.1.4	selectionSort	8
2.9	src/ordenacion.h File Reference	8
2.9.1	Function Documentation	8
2.9.1.1	bubbleSort	8
2.9.1.2	insertionSort	8
2.9.1.3	quickSort	8
2.9.1.4	selectionSort	8
2.10	src/test.cpp File Reference	8
2.10.1	Function Documentation	9
2.10.1.1	generateArray	9
2.10.1.2	makeFourCopies	9
2.11	src/test.h File Reference	9
2.11.1	Function Documentation	9
2.11.1.1	generateArray	9
2.11.1.2	makeFourCopies	9

Chapter 1

File Index

1.1 File List

Here is a list of all files with brief descriptions:

src/ aux.cpp	3
src/ aux.h	4
src/ generators.cpp	4
src/ generators.h	5
src/ interaction.cpp	6
src/ interaction.h	7
src/ main.cpp	7
src/ ordenacion.cpp	7
src/ ordenacion.h	8
src/ test.cpp	8
src/ test.h	9

Chapter 2

File Documentation

2.1 src/aux.cpp File Reference

```
#include "aux.h"
```

Functions

- void **createRandomArray** (int V[], int n, int maxInteger)
Creates a random integer array between 0 and.
- void **copyArray** (int V[], int W[], int n)
- bool **checkSameLength** (int V[], int W[])
- bool **checkArrayEquality** (int V[], int W[], int n)
- bool **checkIsOrdered** (int V[], int n)
- void **printArrayContents** (int V[], int n)

2.1.1 Function Documentation

2.1.1.1 bool checkArrayEquality (int V[], int W[], int n)

2.1.1.2 bool checkIsOrdered (int V[], int n)

2.1.1.3 bool checkSameLength (int V[], int W[])

2.1.1.4 void copyArray (int V[], int W[], int n)

2.1.1.5 void createRandomArray (int V[], int n, int *maxInteger*)

Creates a random integer array between 0 and.

2.1.1.6 void printArrayContents (int V[], int n)

2.2 src/aux.h File Reference

```
#include <time.h> #include <cstdlib> #include <iostream> ×
#include "ordenacion.h" #include <fstream> #include <iomanip> ×
```

Functions

- void **createRandomArray** (int V[], int arrayLegth, int maxInteger)
Creates a random integer array between 0 and.
- void **copyArray** (int V[], int W[], int n)
- bool **checkArrayEquality** (int V[], int W[], int n)
- bool **checkIsOrdered** (int V[], int n)
- void **printArrayContents** (int V[], int n)

2.2.1 Function Documentation

2.2.1.1 bool checkArrayEquality (int V[], int W[], int n)

2.2.1.2 bool checkIsOrdered (int V[], int n)

2.2.1.3 void copyArray (int V[], int W[], int n)

2.2.1.4 void createRandomArray (int V[], int arrayLegth, int maxInteger)

Creates a random integer array between 0 and.

2.2.1.5 void printArrayContents (int V[], int n)

2.3 src/generators.cpp File Reference

```
#include "generators.h"
```

Functions

- void **generateInsertionSortTime** (ofstream &file, int V[], int n)
- void **generateSelectionSortTime** (ofstream &file, int V[], int n)
- void **generateBubbleSortTime** (ofstream &file, int V[], int n)
- void **generateQuickSortTime** (ofstream &file, int V[], int n)
- void **generateInsertionSortDataFile** (int problemSize, int V[], int GAP)
- void **generateSelectionSortDataFile** (int problemSize, int V[], int GAP)

- void **generateBubbleSortDataFile** (int problemSize, int V[], int GAP)
- void **generateQuickSortDataFile** (int problemSize, int V[], int GAP)
- void **generateAllFiles** (int problemSize, int V[], int GAP)

2.3.1 Function Documentation

2.3.1.1 void **generateAllFiles** (int *problemSize*, int *V[]*, int *GAP*)

2.3.1.2 void **generateBubbleSortDataFile** (int *problemSize*, int *V[]*, int *GAP*)

2.3.1.3 void **generateBubbleSortTime** (ofstream & *file*, int *V[]*, int *n*)

2.3.1.4 void **generateInsertionSortDataFile** (int *problemSize*, int *V[]*, int *GAP*)

2.3.1.5 void **generateInsertionSortTime** (ofstream & *file*, int *V[]*, int *n*)

2.3.1.6 void **generateQuickSortDataFile** (int *problemSize*, int *V[]*, int *GAP*)

2.3.1.7 void **generateQuickSortTime** (ofstream & *file*, int *V[]*, int *n*)

2.3.1.8 void **generateSelectionSortDataFile** (int *problemSize*, int *V[]*, int *GAP*)

2.3.1.9 void **generateSelectionSortTime** (ofstream & *file*, int *V[]*, int *n*)

2.4 src/generators.h File Reference

```
#include "aux.h"
```

Functions

- void **generateInsertionSortTime** (ofstream &file, int V[], int n)
- void **generateSelectionSortTime** (ofstream &file, int V[], int n)
- void **generateBubbleSortTime** (ofstream &file, int V[], int n)
- void **generateQuickSortTime** (ofstream &file, int V[], int n)
- void **generateInsertionSortDataFile** (int problemSize, int V[], int GAP)
- void **generateSelectionSortDataFile** (int problemSize, int V[], int GAP)
- void **generateBubbleSortDataFile** (int problemSize, int V[], int GAP)
- void **generateQuickSortDataFile** (int problemSize, int V[], int GAP)
- void **generateAllFiles** (int problemSize, int V[], int GAP)

2.4.1 Function Documentation

2.4.1.1 void **generateAllFiles** (int *problemSize*, int *V[]*, int *GAP*)

2.4.1.2 void generateBubbleSortDataFile (int *problemSize*, int *V[]*, int *GAP*)

2.4.1.3 void generateBubbleSortTime (ofstream & *file*, int *V[]*, int *n*)

2.4.1.4 void generateInsertionSortDataFile (int *problemSize*, int *V[]*, int *GAP*)

2.4.1.5 void generateInsertionSortTime (ofstream & *file*, int *V[]*, int *n*)

2.4.1.6 void generateQuickSortDataFile (int *problemSize*, int *V[]*, int *GAP*)

2.4.1.7 void generateQuickSortTime (ofstream & *file*, int *V[]*, int *n*)

2.4.1.8 void generateSelectionSortDataFile (int *problemSize*, int *V[]*, int *GAP*)

2.4.1.9 void generateSelectionSortTime (ofstream & *file*, int *V[]*, int *n*)

2.5 src/interaction.cpp File Reference

```
#include "interaction.h" #include <iostream>
```

Functions

- int **getProblemSize** ()
Sets problem's size (random array's size)
- int **getMaximumInteger** ()
- int **getGap** ()
- char **chooseAlgorithm** ()

2.5.1 Function Documentation

2.5.1.1 char chooseAlgorithm ()

2.5.1.2 int getGap ()

2.5.1.3 int getMaximumInteger ()

2.5.1.4 int getProblemSize ()

Sets problem's size (random array's size)

Returns

a positive integer that will represent problem's size hereinafter

2.6 src/interaction.h File Reference

Functions

- int **getProblemSize** ()
Sets problem's size (random array's size)
- int **getMaximumInteger** ()
- int **getGap** ()
- char **chooseAlgorithm** ()

2.6.1 Function Documentation

2.6.1.1 char **chooseAlgorithm** ()

2.6.1.2 int **getGap** ()

2.6.1.3 int **getMaximumInteger** ()

2.6.1.4 int **getProblemSize** ()

Sets problem's size (random array's size)

Returns

a positive integer that will represent problem's size hereinafter

2.7 src/main.cpp File Reference

```
#include "test.h" #include "interaction.h" #include "generators.-  
h"
```

Functions

- int **main** ()

2.7.1 Function Documentation

2.7.1.1 int **main** ()

2.8 src/ordenacion.cpp File Reference

```
#include "ordenacion.h"
```

Functions

- void **insertionSort** (int V[], int num)
- void **selectionSort** (int V[], int num)
- void **bubbleSort** (int V[], int num)
- void **quickSort** (int V[], int left, int right)

2.8.1 Function Documentation

2.8.1.1 void bubbleSort (int V[], int num)

2.8.1.2 void insertionSort (int V[], int num)

2.8.1.3 void quickSort (int V[], int left, int right)

2.8.1.4 void selectionSort (int V[], int num)

2.9 src/ordenacion.h File Reference

Functions

- void **insertionSort** (int V[], int num)
- void **selectionSort** (int V[], int num)
- void **bubbleSort** (int V[], int num)
- void **quickSort** (int V[], int left, int right)

2.9.1 Function Documentation

2.9.1.1 void bubbleSort (int V[], int num)

2.9.1.2 void insertionSort (int V[], int num)

2.9.1.3 void quickSort (int V[], int left, int right)

2.9.1.4 void selectionSort (int V[], int num)

2.10 src/test.cpp File Reference

```
#include "test.h"
```

Functions

- void **makeFourCopies** (int V[], int first[], int second[], int third[], int fourth[], int n)
- void **generateArray** (int V[], int n, int maxInt)
Generates an array with n random integers between 0 and maxInt.

2.10.1 Function Documentation

2.10.1.1 void generateArray (int *V*[], int *n*, int *maxInt*)

Generates an array with n random integers between 0 and maxInt.

Parameters

<i>V</i>	the container array with a size of n
<i>n</i>	The array length
<i>maxInt</i>	maximum integer to generate array to

2.10.1.2 void makeFourCopies (int *V*[], int *first*[], int *second*[], int *third*[], int *fourth*[], int *n*)

2.11 src/test.h File Reference

```
#include "aux.h"
```

Functions

- void **generateArray** (int *V*[], int *n*, int *maxInt*)
Generates an array with n random integers between 0 and maxInt.
- void **makeFourCopies** (int *V*[], int *first*[], int *second*[], int *third*[], int *fourth*[], int *n*)

2.11.1 Function Documentation

2.11.1.1 void generateArray (int *V*[], int *n*, int *maxInt*)

Generates an array with n random integers between 0 and maxInt.

Parameters

<i>V</i>	the container array with a size of n
<i>n</i>	The array length
<i>maxInt</i>	maximum integer to generate array to

2.11.1.2 void makeFourCopies (int *V*[], int *first*[], int *second*[], int *third*[], int *fourth*[], int *n*)