

Lab 13: Performance Evaluation

Generated by Doxygen 1.8.5

Fri Oct 4 2013 01:58:15

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	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	binarySearch Class Reference	7
4.1.1	Member Function Documentation	7
4.1.1.1	operator()	7
4.2	linearSearch Class Reference	7
4.2.1	Member Function Documentation	8
4.2.1.1	operator()	8
4.3	Search Class Reference	8
4.4	STLSearch Class Reference	8
4.4.1	Member Function Documentation	8
4.4.1.1	operator()	8
4.5	TestVector Class Reference	9
4.5.1	Constructor & Destructor Documentation	9
4.5.1.1	TestVector	9
4.5.1.2	TestVector	9
4.5.2	Member Function Documentation	9
4.5.2.1	operator++	9
4.5.2.2	operator++	9
4.5.2.3	operator[]	9
4.6	Timer Class Reference	9
4.6.1	Constructor & Destructor Documentation	9
4.6.1.1	Timer	9
4.6.2	Member Function Documentation	10

4.6.2.1	getElapsedTime	10
4.6.2.2	start	10
4.6.2.3	stop	11
5	File Documentation	13
5.1	config.h File Reference	13
5.1.1	Macro Definition Documentation	13
5.1.1.1	LAB13_TEST1	13
5.1.1.2	LAB13_TEST2	13
5.2	constructor.cpp File Reference	13
5.2.1	Macro Definition Documentation	14
5.2.1.1	runTest	14
5.2.2	Function Documentation	14
5.2.2.1	main	14
5.2.2.2	testCompute	14
5.2.2.3	testCompute< double >	14
5.2.2.4	testCompute< int >	14
5.2.2.5	testConstructor	14
5.2.3	Variable Documentation	14
5.2.3.1	numRepetitions	14
5.3	inc.cpp File Reference	14
5.3.1	Function Documentation	14
5.3.1.1	main	14
5.3.2	Variable Documentation	14
5.3.2.1	numRepetitions	14
5.4	search.cpp File Reference	15
5.4.1	Function Documentation	15
5.4.1.1	main	15
5.4.2	Variable Documentation	15
5.4.2.1	numSearches	15
5.5	sort.cpp File Reference	15
5.5.1	Function Documentation	16
5.5.1.1	main	16
5.5.1.2	quickSort	16
5.5.1.3	selectionSort	16
5.5.1.4	timeSort	16
5.5.2	Variable Documentation	16
5.5.2.1	numSorts	16
5.6	test.cpp File Reference	16
5.6.1	Function Documentation	16

5.6.1.1	main	16
5.7	test13.cpp File Reference	16
5.7.1	Function Documentation	16
5.7.1.1	main	16
5.7.1.2	print_help	16
5.7.1.3	wait	16
5.8	testtimer.c++ File Reference	17
5.8.1	Function Documentation	17
5.8.1.1	main	17
5.9	testtimer.cc File Reference	17
5.9.1	Function Documentation	17
5.9.1.1	main	17
5.10	testtimer.cpp File Reference	17
5.10.1	Function Documentation	17
5.10.1.1	main	17
5.11	testvector.cpp File Reference	18
5.12	testvector.h File Reference	18
5.13	text.cc File Reference	18
5.13.1	Function Documentation	18
5.13.1.1	main	18
5.14	Timer.cpp File Reference	18
5.14.1	Macro Definition Documentation	19
5.14.1.1	TIMER_CPP	19
5.14.2	Function Documentation	19
5.14.2.1	toddiff	19
5.15	Timer.cs File Reference	19
5.15.1	Macro Definition Documentation	19
5.15.1.1	TIMER_CPP	19
5.16	Timer.h File Reference	19

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

binary_function	
Search	8
binarySearch	7
linearSearch	7
STLSearch	8
TestVector	9
Timer	9

Chapter 2

Class Index

2.1 Class List

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

binarySearch	7
linearSearch	7
Search	8
STLSearch	8
TestVector	9
Timer	9

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

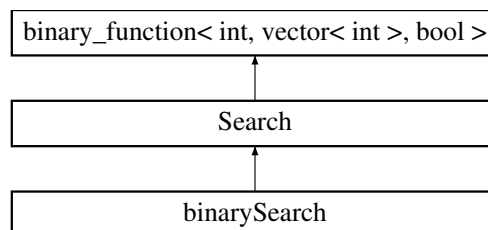
config.h	13
constructor.cpp	13
inc.cpp	14
search.cpp	15
sort.cpp	15
test.cpp	16
test13.cpp	16
testtimer.c++	17
testtimer.cc	17
testtimer.cpp	17
testvector.cpp	18
testvector.h	18
text.cc	18
Timer.cpp	18
Timer.cs	19
Timer.h	19

Chapter 4

Class Documentation

4.1 binarySearch Class Reference

Inheritance diagram for binarySearch:



Public Member Functions

- `bool operator\(\) (int searchValue, const vector< int > &keys) const`

4.1.1 Member Function Documentation

4.1.1.1 `bool binarySearch::operator() (int searchValue, const vector< int > & keys) const` `[inline],[virtual]`

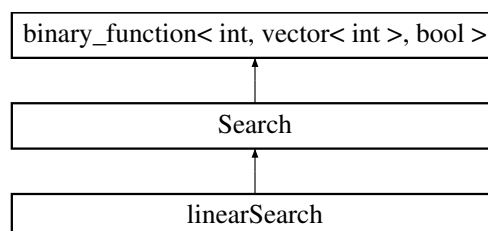
Implements [Search](#).

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

- [search.cpp](#)

4.2 linearSearch Class Reference

Inheritance diagram for linearSearch:



Public Member Functions

- `bool operator() (int searchValue, const vector< int > &keys) const`

4.2.1 Member Function Documentation

4.2.1.1 `bool linearSearch::operator() (int searchValue, const vector< int > & keys) const` `[inline],[virtual]`

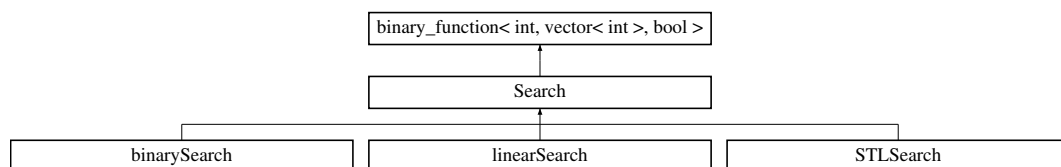
Implements [Search](#).

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

- [search.cpp](#)

4.3 Search Class Reference

Inheritance diagram for Search:

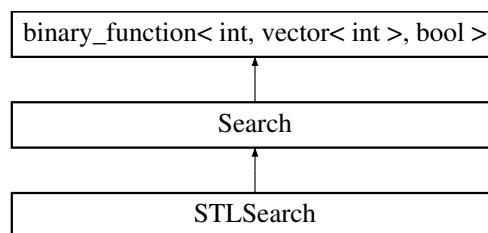


The documentation for this class was generated from the following file:

- [search.cpp](#)

4.4 STLSearch Class Reference

Inheritance diagram for STLSearch:



Public Member Functions

- `bool operator() (int searchValue, const vector< int > &keys) const`

4.4.1 Member Function Documentation

4.4.1.1 `bool STLSearch::operator() (int searchValue, const vector< int > & keys) const` `[inline],[virtual]`

Implements [Search](#).

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

- [search.cpp](#)

4.5 TestVector Class Reference

```
#include <testvector.h>
```

Public Member Functions

- [TestVector](#) (int size)
- [TestVector](#) (const [TestVector](#) &rhs)
- [TestVector](#) & [operator++](#) ()
- [TestVector](#) [operator++](#) (int ignored)
- int [operator\[\]](#) (int loc) const

4.5.1 Constructor & Destructor Documentation

4.5.1.1 [TestVector::TestVector](#) (int *size*)

4.5.1.2 [TestVector::TestVector](#) (const [TestVector](#) & *rhs*)

4.5.2 Member Function Documentation

4.5.2.1 [TestVector](#) & [TestVector::operator++](#) ()

4.5.2.2 [TestVector](#) [TestVector::operator++](#) (int *ignored*)

4.5.2.3 int [TestVector::operator\[\]](#) (int *loc*) const

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

- [testvector.h](#)
- [testvector.cpp](#)

4.6 Timer Class Reference

```
#include <Timer.h>
```

Public Member Functions

- [Timer](#) ()
- void [start](#) () throw (runtime_error)
- void [stop](#) () throw (logic_error)
- double [getElapsedTime](#) () const throw (logic_error)

4.6.1 Constructor & Destructor Documentation

4.6.1.1 [Timer::Timer](#) ()

Precondition

Unitialized `Timer` Class

Postcondition

Duration assigned to -1 timerWasStarted assigned false

Algorithm:

- Assign data members values

Exceptional/Error Conditions:

- none

4.6.2 Member Function Documentation**4.6.2.1 `double Timer::getElapsedTime () const throw logic_error`**

Parameters: none

Returns

The time in seconds as a double

Precondition

none

Postcondition

Duration set to the number of milliseconds passed since the start function was called

Algorithm:

- Get time of day assigned to the begin time
- Assign timerWasStarted true

Exceptions

<i>logic_error</i>	A logic error is thrown if the start function was never called or if the stop function was never called.
--------------------	--

4.6.2.2 `void Timer::start () throw runtime_error`**Precondition**

none

Postcondition

beginTime set to the timeval timerWasStarted assigned true

Algorithm:

- Get time of day assigned to the begin time
- Assign timerWasStarted true

Exceptional/Error Conditions:

- none

4.6.2.3 void Timer::stop () throw logic_error)**Precondition**

none

Postcondition

Duration set to the number of milliseconds passed since the start function was called

Algorithm:

- Get time of day assigned to the begin time
- Assign timerWasStarted true

Exceptions

A	logic error is thrown if the start function was never called.
---	---

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

- [Timer.h](#)
- [Timer.cpp](#)
- [Timer.cs](#)

Chapter 5

File Documentation

5.1 config.h File Reference

Macros

- `#define LAB13_TEST1 0`
- `#define LAB13_TEST2 0`

5.1.1 Macro Definition Documentation

5.1.1.1 `#define LAB13_TEST1 0`

`Timer` class (Lab 13) configuration file. Activate test 'N' by defining the corresponding LAB12_TESTN to have the value 1.

5.1.1.2 `#define LAB13_TEST2 0`

5.2 constructor.cpp File Reference

```
#include <iostream>
#include <string>
#include "Timer.h"
#include "TestVector.h"
```

Macros

- `#define runTest(Type) testConstructor<Type>(numValues, #Type)`

Functions

- `template<typename DataType >`
`int testCompute (DataType value)`
- `template<>`
`int testCompute< int > (int value)`
- `template<>`
`int testCompute< double > (double value)`

- `template<typename DataType >`
`void testConstructor (int numValues, string name)`
- `int main (int argc, char **argv)`

Variables

- `const int numRepetitions = 1000000`

5.2.1 Macro Definition Documentation

5.2.1.1 `#define runTest(Type) testConstructor<Type>(numValues, #Type)`

5.2.2 Function Documentation

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

5.2.2.2 `template<typename DataType > int testCompute (DataType value)`

5.2.2.3 `template<> int testCompute< double > (double value)`

5.2.2.4 `template<> int testCompute< int > (int value)`

5.2.2.5 `template<typename DataType > void testConstructor (int numValues, string name)`

5.2.3 Variable Documentation

5.2.3.1 `const int numRepetitions = 1000000`

5.3 inc.cpp File Reference

```
#include <iostream>
#include "Timer.h"
#include "TestVector.h"
```

Functions

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

Variables

- `const int numRepetitions = 1000000`

5.3.1 Function Documentation

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

5.3.2 Variable Documentation

5.3.2.1 `const int numRepetitions = 1000000`

5.4 search.cpp File Reference

```
#include <iostream>
#include <algorithm>
#include <vector>
#include "Timer.h"
```

Classes

- class [Search](#)
- class [linearSearch](#)
- class [binarySearch](#)
- class [STLSearch](#)

Functions

- int [main](#) (int argc, char **argv)

Variables

- const int [numSearches](#) = 100000

5.4.1 Function Documentation

5.4.1.1 int main (int *argc*, char ** *argv*)

5.4.2 Variable Documentation

5.4.2.1 const int numSearches = 100000

5.5 sort.cpp File Reference

```
#include <iostream>
#include <algorithm>
#include <vector>
#include "Timer.h"
```

Functions

- void [selectionSort](#) (vector< int >::iterator front, vector< int >::iterator back)
- void [quickSort](#) (vector< int >::iterator front, vector< int >::iterator back)
- void [timeSort](#) (void(*fcn)(vector< int >::iterator front, vector< int >::iterator back), const string name, const vector< int > &masterList, const [Timer](#) &overhead)
- int [main](#) (int argc, char **argv)

Variables

- const int [numSorts](#) = 100

5.5.1 Function Documentation

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

5.5.1.2 `void quickSort (vector< int >::iterator front, vector< int >::iterator back)`

5.5.1.3 `void selectionSort (vector< int >::iterator front, vector< int >::iterator back)`

5.5.1.4 `void timeSort (void(*)(vector< int >::iterator front, vector< int >::iterator back) fcn, const string name, const vector< int > & masterList, const Timer & overhead)`

5.5.2 Variable Documentation

5.5.2.1 `const int numSorts = 100`

5.6 test.cpp File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
```

Functions

- `int main ()`

5.6.1 Function Documentation

5.6.1.1 `int main ()`

5.7 test13.cpp File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
#include "Timer.h"
```

Functions

- `void wait (int secs)`
- `void print_help ()`
- `int main ()`

5.7.1 Function Documentation

5.7.1.1 `int main ()`

5.7.1.2 `void print_help ()`

5.7.1.3 `void wait (int secs)`

5.8 testtimer.c++ File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
#include "Timer.h"
```

Functions

- int [main](#) ()

5.8.1 Function Documentation

5.8.1.1 int main ()

5.9 testtimer.cc File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
#include "Timer.h"
```

Functions

- int [main](#) ()

5.9.1 Function Documentation

5.9.1.1 int main ()

5.10 testtimer.cpp File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
#include <iomanip>
#include "Timer.h"
```

Functions

- int [main](#) ()

5.10.1 Function Documentation

5.10.1.1 int main ()

5.11 testvector.cpp File Reference

```
#include <functional>
#include <algorithm>
#include "TestVector.h"
```

5.12 testvector.h File Reference

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

Classes

- class [TestVector](#)

5.13 text.cc File Reference

```
#include <iostream>
#include <cctype>
#include <ctime>
```

Functions

- int [main](#) ()

5.13.1 Function Documentation

5.13.1.1 int main ()

5.14 Timer.cpp File Reference

```
#include <iostream>
#include <sys/time.h>
#include <Timer.h>
```

Macros

- #define [TIMER_CPP](#)

Functions

- long long int [toddiff](#) (struct timeval *tod1, struct timeval *tod2)

5.14.1 Macro Definition Documentation

5.14.1.1 #define TIMER_CPP

5.14.2 Function Documentation

5.14.2.1 long long int toddiff (struct timeval * *tod1*, struct timeval * *tod2*)

Parameters

<i>tod1</i>	This is the initial timeval
<i>tod2</i>	This is the final timeval

Precondition

none

Postcondition

The difference between the intial and final time is returned.

Returns

Returns the difference between the intial and final time in usec.

Algorithm:

- Converts the timevals to be measured in usecs
- Returns the difference between the two times in usecs

Exceptional/Error Conditions:

- none

5.15 Timer.cs File Reference

```
#include "Timer.h"
```

Macros

- #define [TIMER_CPP](#)

5.15.1 Macro Definition Documentation

5.15.1.1 #define TIMER_CPP

5.16 Timer.h File Reference

```
#include <sys/time.h>
#include <stdexcept>
#include <iostream>
```

Classes

- class [Timer](#)

Index

- binarySearch, 7
 - operator(), 7
- config.h, 13
 - LAB13_TEST1, 13
 - LAB13_TEST2, 13
- constructor.cpp, 13
 - main, 14
 - numRepetitions, 14
 - runTest, 14
 - testCompute, 14
 - testCompute< double >, 14
 - testCompute< int >, 14
 - testConstructor, 14
- getElapsedTime
 - Timer, 10
- inc.cpp, 14
 - main, 14
 - numRepetitions, 14
- LAB13_TEST1
 - config.h, 13
- LAB13_TEST2
 - config.h, 13
- linearSearch, 7
 - operator(), 8
- main
 - constructor.cpp, 14
 - inc.cpp, 14
 - search.cpp, 15
 - sort.cpp, 16
 - test.cpp, 16
 - test13.cpp, 16
 - testtimer.c++, 17
 - testtimer.cc, 17
 - testtimer.cpp, 17
 - text.cc, 18
- numRepetitions
 - constructor.cpp, 14
 - inc.cpp, 14
- numSearches
 - search.cpp, 15
- numSorts
 - sort.cpp, 16
- operator()
 - binarySearch, 7
 - linearSearch, 8
 - STLSearch, 8
- operator++
 - TestVector, 9
- print_help
 - test13.cpp, 16
- quickSort
 - sort.cpp, 16
- runTest
 - constructor.cpp, 14
- STLSearch, 8
 - operator(), 8
- Search, 8
- search.cpp, 15
 - main, 15
 - numSearches, 15
- selectionSort
 - sort.cpp, 16
- sort.cpp, 15
 - main, 16
 - numSorts, 16
 - quickSort, 16
 - selectionSort, 16
 - timeSort, 16
- start
 - Timer, 10
- stop
 - Timer, 11
- TIMER_CPP
 - Timer.cpp, 19
 - Timer.cs, 19
- test.cpp, 16
 - main, 16
- test13.cpp, 16
 - main, 16
 - print_help, 16
 - wait, 16
- testCompute
 - constructor.cpp, 14
- testCompute< double >
 - constructor.cpp, 14
- testCompute< int >
 - constructor.cpp, 14
- testConstructor
 - constructor.cpp, 14
- TestVector, 9

- operator++, [9](#)
 - TestVector, [9](#)
 - TestVector, [9](#)
- testtimer.c++, [17](#)
 - main, [17](#)
- testtimer.cc, [17](#)
 - main, [17](#)
- testtimer.cpp, [17](#)
 - main, [17](#)
- testvector.cpp, [18](#)
- testvector.h, [18](#)
- text.cc, [18](#)
 - main, [18](#)
- timeSort
 - sort.cpp, [16](#)
- Timer, [9](#)
 - getElapsedTime, [10](#)
 - start, [10](#)
 - stop, [11](#)
 - Timer, [9](#)
- Timer.cpp, [18](#)
 - TIMER_CPP, [19](#)
 - toddiff, [19](#)
- Timer.cs, [19](#)
 - TIMER_CPP, [19](#)
- Timer.h, [19](#)
- toddiff
 - Timer.cpp, [19](#)
- wait
 - test13.cpp, [16](#)