Documentation for _vec3 Class and Unit Tests

Generated by Doxygen 1.9.2

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 _vec3< T > Class Template Reference	7
4.1.1 Detailed Description	7
4.1.2 Constructor & Destructor Documentation	8
4.1.2.1 _vec3() [1/2]	8
4.1.2.2 _vec3() [2/2]	8
4.1.3 Member Function Documentation	8
4.1.3.1 operator[]() [1/2]	8
4.1.3.2 operator[]() [2/2]	8
4.1.4 Friends And Related Function Documentation	10
4.1.4.1 dot	10
4.1.4.2 mag	10
4.1.4.3 operator+	11
4.1.4.4 operator	11
4.1.4.5 operator <<	11
4.1.5 Member Data Documentation	12
4.1.5.1 d	12
4.2 VecTestClass Class Reference	12
4.2.1 Detailed Description	13
4.2.2 Constructor & Destructor Documentation	13
4.2.2.1 VecTestClass()	13
4.2.2.2 ~VecTestClass()	13
4.2.3 Member Function Documentation	13
4.2.3.1 SetUp()	13
4.2.3.2 TearDown()	13
4.2.4 Member Data Documentation	13
4.2.4.1 a	14
4.2.4.2 add_matches	14
4.2.4.3 b	14
4.2.4.4 sub_matches	14
5 File Documentation	15
5.1 Vec.h File Reference	15
5.1.1 Typedef Documentation	15
5.1.1.1 double3	15

	5.2 Vec.h	16
	5.3 Vec_tests.cpp File Reference	16
	5.3.1 Function Documentation	17
	5.3.1.1 main()	17
	5.3.1.2 TEST()	17
	5.3.1.3 TEST_F() [1/2]	17
	5.3.1.4 TEST_F() [2/2]	18
		40
Inde	ex	19

Hierarchical Index

1.1 Class Hierarchy

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

$_{\text{vec3}}$ < T >								 	 													7
_vec3< double >								 	 													7
testing::Test																						
VecTestClass		 		 											 							12

2 Hierarchical Index

Class Index

2.1 Class List

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

$_{\text{vec3}} < 1 > \dots \dots$	
VecTestClass	
This is a test class	

4 Class Index

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Vec.h	. 15
Vec tests.cpp	. 16

6 File Index

Class Documentation

4.1 _vec3< T > Class Template Reference

```
#include <Vec.h>
```

Public Member Functions

- _vec3 ()
- _vec3 (T a, T b, T c)
- T & operator[] (int i)
- T operator[] (int i) const

Protected Attributes

• T d [3]

Friends

```
• _{\text{vec3}}< T > operator+ (const _{\text{vec3}}< T > &a, const _{\text{vec3}}< T > &b)
```

- _vec3< T > operator- (const _vec3< T > &a, const _vec3< T > &b)
- T dot (const _vec3< T > &a, const _vec3< T > &b)
- double mag (const _vec3< T > &a)
- std::ostream & operator<< (std::ostream &out, const _vec3< T > &a)

4.1.1 Detailed Description

```
\label{eq:typename} \begin{array}{l} \text{template}\!<\!\text{typename T}\!>\\ \text{class}\,\_\text{vec3}\!<\!\text{T}\!> \end{array}
```

A class for creating a generic three-dimensional vector of type T. Includes operator overloading (for +, -, [], and <<) so that vector operations can be performed.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 _vec3() [1/2]

```
template<typename T >
_vec3< T >::_vec3 () [inline]
```

Constructor

4.1.2.2 _vec3() [2/2]

Destructor

4.1.3 Member Function Documentation

4.1.3.1 operator[]() [1/2]

Operator for assigning a value to a component of the vector.

Parameters

```
in i The index of the component to be accessed.
```

Returns

Reference to the i-th component of the vector.

4.1.3.2 operator[]() [2/2]

```
template<typename T > T _vec3< T >::operator[] ( int i ) const [inline]
```

Operator for reading a component of the vector.

Parameters

in	i	The index of the component to be accessed.
----	---	--

Returns

The i-th component of the vector.

4.1.4 Friends And Related Function Documentation

4.1.4.1 dot

Performs dot product of two vectors.

Parameters

in	а	Vector to be dotted.
in	b	Vector to be dotted.

Returns

 $a \cdot b$

4.1.4.2 mag

Computes magnitude of a vector.

Parameters

in	а	Vector with unknown magnitude.
----	---	--------------------------------

Returns

 $\sqrt{a \cdot b}$

4.1.4.3 operator+

```
template<typename T > _vec3< T > operator+ (  const \_vec3< T > \& a, \\ const \_vec3< T > \& b ) [friend]
```

Operator for adding two vectors together.

Parameters

in	а	Vector to be added.
in	b	Vector to be added.

Returns

a + b

4.1.4.4 operator-

```
template<typename T > _vec3< T > operator- (  const \_vec3< T > \& a, \\ const \_vec3< T > \& b ) [friend]
```

Operator for subtracting one vector from another.

Parameters

in	а	Vector to be subtracted from.
in	b	Vector to be subtracted.

Returns

a - b

4.1.4.5 operator <<

Operator for outputting vector components to a stream.

Parameters

in	out	Output stream object.
in	а	Vector to be outputted to stream.

Returns

Output stream object, so that operator can be used several times in a row.

4.1.5 Member Data Documentation

4.1.5.1 d

```
template<typename T >
T _vec3< T >::d[3] [protected]
```

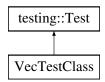
The documentation for this class was generated from the following file:

Vec.h

4.2 VecTestClass Class Reference

This is a test class.

Inheritance diagram for VecTestClass:



Protected Member Functions

- VecTestClass ()
- \sim VecTestClass () override
- void SetUp () override
- void TearDown () override

Protected Attributes

- · double3 a
- double3 b
- bool sub_matches = false
- bool add_matches = false

4.2.1 Detailed Description

This is a test class.

It will be used to test the addition and subtraction operators of the <u>vec3</u><double> class. GTest will automatically call the constructor and the SetUp function prior to each test, and TearDown and the destructor afterwards.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 VecTestClass()

```
VecTestClass::VecTestClass ( ) [inline], [protected]
```

Constructor

4.2.2.2 ∼VecTestClass()

```
VecTestClass::~VecTestClass ( ) [inline], [override], [protected]
```

Destructor

4.2.3 Member Function Documentation

4.2.3.1 SetUp()

```
void VecTestClass::SetUp ( ) [inline], [override], [protected]
```

Performs addition/subtraction of double3 objects, and compares with the direct addition/subtraction of double3 elements.

4.2.3.2 TearDown()

```
void VecTestClass::TearDown ( ) [inline], [override], [protected]
```

Optional code to call after each test prior to destructor.

4.2.4 Member Data Documentation

4.2.4.1 a

```
double3 VecTestClass::a [protected]
```

This is a vec3_ object, as defined in Vec.h, and it uses double as template

4.2.4.2 add_matches

```
bool VecTestClass::add_matches = false [protected]
```

Does the addition result match our expectation?

4.2.4.3 b

```
double3 VecTestClass::b [protected]
```

This is a vec3_ object, as defined in Vec.h, and it uses double as template

4.2.4.4 sub_matches

```
bool VecTestClass::sub_matches = false [protected]
```

Does the subtraction result match our expectation?

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

• Vec_tests.cpp

File Documentation

5.1 Vec.h File Reference

```
#include <ostream>
#include <math.h>
```

Classes

class _vec3< T >

Typedefs

• using double3 = _vec3< double >

5.1.1 Typedef Documentation

5.1.1.1 double3

using double3 = _vec3<double>

16 File Documentation

5.2 Vec.h

Go to the documentation of this file.

```
1 #ifndef _VEC_H
2 #define _VEC_H
3 #include <ostream>
4 #include <math.h>
11 template<typename T>
12 class _vec3{
13 public:
       _vec3<T>(): d{0,0,0} {}
18
19
23
       _vec3<T>(T a, T b, T c) : d{a,b,c} {}
       T& operator[] (int i) {return d[i];}
30
31
       T operator[] (int i) const {return d[i];}
45
       friend _vec3<T> operator+(const _vec3<T>&a, const _vec3<T>&b) {
46
           return _vec3<T>(a[0]+b[0],a[1]+b[1],a[2]+b[2]);
47
48
55
       friend _vec3<T> operator-(const _vec3<T>&a, const _vec3<T>&b) {
           return _vec3<T>(a[0]-b[0],a[1]-b[1],a[2]-b[2]);
58
      friend T dot(const _vec3<T>&a, const _vec3<T>&b) {
6.5
            return a[0]*b[0]+a[1]*b[1]+a[2]*b[2];
66
68
       friend double mag(const _vec3<T>&a) {return sqrt(dot(a,a));}
75
       friend std::ostream& operator«(std::ostream &out, const _vec3<T>&a) {
   out«a[0]«" "«a[1]«" "«a[2]; return out;
82
83
84
85
       T d[3]; };
89 using double3 = _vec3<double>;
90
91 #endif
```

5.3 Vec_tests.cpp File Reference

```
#include "gtest/gtest.h"
#include <iostream>
#include "Vec.h"
```

Classes

class VecTestClass

This is a test class.

Functions

• TEST (VecTest, VecDot)

Tests the dot() function of the _vec3 class.

- TEST_F (VecTestClass, VecAdd)
- TEST_F (VecTestClass, VecSub)
- int main (int argc, char **argv)

5.3.1 Function Documentation

5.3.1.1 main()

```
int main (  \mbox{int $argc$,} \\ \mbox{char $**$ $argv$ )}
```

Runs several unit tests through GTest

Parameters

in	argc	Argument count - number of inputs to command line at runtime
in	argv	Argument vector - an array of character pointers listing all command line arguments

Returns

A call to run all GTest tests

5.3.1.2 TEST()

```
TEST (

VecTest ,

VecDot )
```

Tests the dot() function of the _vec3 class.

5.3.1.3 TEST_F() [1/2]

Uses the VecTestClass class to test the + operator of the _vec3 class

Parameters

in	VectTestClass	Class for testing the _vec3 class

18 File Documentation

5.3.1.4 TEST_F() [2/2]

Uses the VecTestClass class to test the - operator of the _vec3 class

Parameters

in	VectTestClass	Class for testing the	_vec3 class	
----	---------------	-----------------------	-------------	--

Index

```
_vec3
     _vec3< T>, 8
_{vec3} < T >, 7
    _vec3, 8
    d, 12
     dot, 10
     mag, 10
     operator <<, 11
     operator+, 11
     operator-, 11
     operator[], 8
\sim\!\text{VecTestClass}
     VecTestClass, 13
а
     VecTestClass, 13
add matches
     VecTestClass, 14
b
     VecTestClass, 14
d
     _{vec3} < T >, 12
dot
     _vec3< T >, 10
double3
     Vec.h, 15
mag
     _{vec3} < T >, 10
main
     Vec_tests.cpp, 17
operator<<
     _{\text{vec3}} < T >, 11
operator+
     _{vec3} < T >, 11
operator-
     _{vec3} < T >, 11
operator[]
     _{vec3} < T >, 8
SetUp
     VecTestClass, 13
sub matches
     VecTestClass, 14
TearDown
     VecTestClass, 13
```

Vec_tests.cpp, 17 TEST_F Vec_tests.cpp, 17 Vec.h, 15 double3, 15 Vec_tests.cpp, 16 main, 17 TEST, 17 TEST_F, 17 VecTestClass, 12 $\sim\!\text{VecTestClass, 13}$ a, 13 add matches, 14 b, 14 SetUp, 13 sub matches, 14 TearDown, 13 VecTestClass, 13

TEST