

aardvark

0.3.0

Generated by Doxygen 1.8.17



---

<b>1 File Index</b>	<b>1</b>
1.1 File List . . . . .	1
<b>2 File Documentation</b>	<b>3</b>
2.1 /home/bona/CPTR227/Sorting/src/main.cpp File Reference . . . . .	3
2.1.1 Detailed Description . . . . .	3
2.1.2 Function Documentation . . . . .	4
2.1.2.1 main() . . . . .	4
2.1.2.2 quickSort() . . . . .	4
<b>Index</b>	<b>5</b>



# Chapter 1

## File Index

### 1.1 File List

Here is a list of all files with brief descriptions:

<code>/home/bona/CPTR227/Sorting/src/<a href="#">main.cpp</a></code>	
This is sorting . . . . .	<a href="#">3</a>



## Chapter 2

# File Documentation

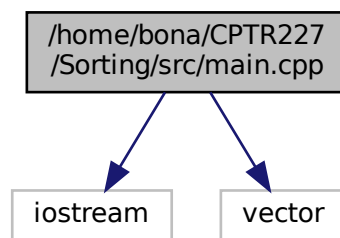
### 2.1 /home/bona/CPTR227/Sorting/src/main.cpp File Reference

This is sorting.

```
#include <iostream>
```

```
#include <vector>
```

Include dependency graph for main.cpp:



### Functions

- void [quickSort](#) (vector< int > &a, int i, int n)
- int [main](#) ()

#### 2.1.1 Detailed Description

This is sorting.

This is the long brief at the top of [main.cpp](#).

Author

Bona Tufa

Date

4/19/2021

## 2.1.2 Function Documentation

### 2.1.2.1 main()

```
int main ( )
```

Definition at line 43 of file main.cpp.

```
43     {
44         vector<int> fag{5, 6, 2, 1, 27, 3};
45         for (int i = 0; i < fag.size(); i++) {
46             std::cout << fag.at(i) << ' ';
47         }
48         quickSort(fag, 0, fag.size());
49         cout << endl;
50         for (int i = 0; i < fag.size(); i++) {
51             std::cout << fag.at(i) << ' ';
52         }
53         cout << "done!";
54     }
```

### 2.1.2.2 quickSort()

```
void quickSort (
    vector< int > & a,
    int i,
    int n )
```

Definition at line 21 of file main.cpp.

```
21     {
22         if (n <= 1) return;
23         int x = a[i + rand()%n];
24         int p = i-1, j = i, q = i+n;
25         while (j < q) {
26             int comp = a[j] - x;
27             if (comp < 0) { // move to beginning of array
28                 swap(a[j++], a[++p]);
29             } else if (comp > 0) {
30                 swap(a[j], a[--q]); // move to end of array
31             } else {
32                 j++; // keep in the middle
33             }
34         }
35
36         quickSort(a, i, p-i+1);
37
38         quickSort(a, q, n-(q-i));
39     }
40 }
```



# Index

[/home/bona/CPTR227/Sorting/src/main.cpp](#), [3](#)

main

[main.cpp](#), [4](#)

[main.cpp](#)

[main](#), [4](#)

[quickSort](#), [4](#)

[quickSort](#)

[main.cpp](#), [4](#)