

sorting

0.1.0

Generated by Doxygen 1.8.17



<b>1 Class Index</b>	<b>1</b>
1.1 Class List	1
<b>2 File Index</b>	<b>3</b>
2.1 File List	3
<b>3 Class Documentation</b>	<b>5</b>
3.1 heapsort Class Reference	5
3.1.1 Detailed Description	5
3.1.2 Member Function Documentation	5
3.1.2.1 BinaryHeap()	6
3.1.2.2 compare()	6
3.1.2.3 left()	6
3.1.2.4 myswap()	6
3.1.2.5 parent()	7
3.1.2.6 print()	7
3.1.2.7 reverse()	7
3.1.2.8 right()	7
3.1.2.9 sort()	7
3.1.2.10 trickleDown()	8
3.1.3 Member Data Documentation	8
3.1.3.1 a	8
3.1.3.2 alength	8
3.1.3.3 n	8
<b>4 File Documentation</b>	<b>9</b>
4.1 /home/brandon/CPTR227/HW15/sorting/src/main.cpp File Reference	9
4.1.1 Detailed Description	10
4.1.2 Function Documentation	10
4.1.2.1 main()	10
<b>Index</b>	<b>11</b>



# Chapter 1

## Class Index

### 1.1 Class List

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

<a href="#">heapsort</a> . . . . .	5
------------------------------------	---



## Chapter 2

# File Index

### 2.1 File List

Here is a list of all files with brief descriptions:

/home/brandon/CPTR227/HW15/sorting/src/ <a href="#">main.cpp</a>	
This is Homework 15-Sorting for CPTR 227 . . . . .	9





## Chapter 3

# Class Documentation

### 3.1 heapsort Class Reference

#### Public Member Functions

- int [left](#) (int i)
- int [right](#) (int i)
- int [parent](#) (int i)
- int [compare](#) (int x, int y)
- void [myswap](#) (int i, int p)
- void [trickleDown](#) (int i)
- void [BinaryHeap](#) ()
- void [reverse](#) ()
- void [sort](#) ()
- void [print](#) ()

#### Public Attributes

- int [a](#) [[alength](#)] = {21,37,426,0,1212,49,814,100,61,3,76,30,705,84,1995,98,12,25,7,55}
- int [n](#) = 20

#### Static Public Attributes

- static const int [alength](#) = 20

#### 3.1.1 Detailed Description

Definition at line 14 of file main.cpp.

#### 3.1.2 Member Function Documentation

### 3.1.2.1 BinaryHeap()

```
void heapsort::BinaryHeap ( ) [inline]
```

Definition at line 68 of file main.cpp.

```
68         {
69             n = a.length;
70             for (int i = n/2-1; i >= 0; i--) {
71                 trickleDown(i);
72             }
73         }
```

### 3.1.2.2 compare()

```
int heapsort::compare (
    int x,
    int y ) [inline]
```

Definition at line 30 of file main.cpp.

```
30         {
31             if (x < y)
32                 return -1;
33             else if (x > y)
34                 return 1;
35             else
36                 return 0;
37         }
```

### 3.1.2.3 left()

```
int heapsort::left (
    int i ) [inline]
```

Definition at line 20 of file main.cpp.

```
20         {
21             return 2*i + 1;
22         }
```

### 3.1.2.4 myswap()

```
void heapsort::myswap (
    int i,
    int p ) [inline]
```

Definition at line 39 of file main.cpp.

```
39         {
40             int temp;
41             temp = a[i];
42             a[i] = a[p];
43             a[p] = temp;
44         }
```

### 3.1.2.5 parent()

```
int heapsort::parent (
    int i ) [inline]
```

Definition at line 26 of file main.cpp.

```
26     {
27         return (i-1)/2;
28     }
```

### 3.1.2.6 print()

```
void heapsort::print ( ) [inline]
```

Definition at line 89 of file main.cpp.

```
89     {
90         for (int i = 0; i < alength; i++)
91             std::cout << a[i] << " ";
92         std::cout << endl;
93     }
```

### 3.1.2.7 reverse()

```
void heapsort::reverse ( ) [inline]
```

Definition at line 75 of file main.cpp.

```
75     {
76         for (int i=0; i < alength/2; i++)
77             myswap(i, (alength-i-1));
78     }
```

### 3.1.2.8 right()

```
int heapsort::right (
    int i ) [inline]
```

Definition at line 23 of file main.cpp.

```
23     {
24         return 2*i + 2;
25     }
```

### 3.1.2.9 sort()

```
void heapsort::sort ( ) [inline]
```

Definition at line 80 of file main.cpp.

```
80     {
81         BinaryHeap();
82         while (n > 1) {
83             myswap(--n, 0);
84             trickleDown(0);
85         }
86         reverse();
87     }
```

### 3.1.2.10 trickleDown()

```
void heapsort::trickleDown (
    int i ) [inline]
```

Definition at line 46 of file main.cpp.

```
46         {
47             do {
48                 int j = -1;
49                 int r = right(i);
50                 if (r < n && compare(a[r], a[i]) < 0) {
51                     int l = left(i);
52                     if (compare(a[l], a[r]) < 0) {
53                         j = l;
54                     } else {
55                         j = r;
56                     }
57                 } else {
58                     int l = left(i);
59                     if (l < n && compare(a[l], a[i]) < 0) {
60                         j = l;
61                     }
62                 }
63                 if (j >= 0) myswap(i, j);
64                 i = j;
65             } while (i >= 0);
66         }
```

## 3.1.3 Member Data Documentation

### 3.1.3.1 a

```
int heapsort::a[alength] = {21,37,426,0,1212,49,814,100,61,3,76,30,705,84,1995,98,12,25,7,55}
```

Definition at line 17 of file main.cpp.

### 3.1.3.2 alength

```
const int heapsort::alength = 20 [static]
```

Definition at line 16 of file main.cpp.

### 3.1.3.3 n

```
int heapsort::n = 20
```

Definition at line 18 of file main.cpp.

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

- </home/brandon/CPTR227/HW15/sorting/src/main.cpp>

## Chapter 4

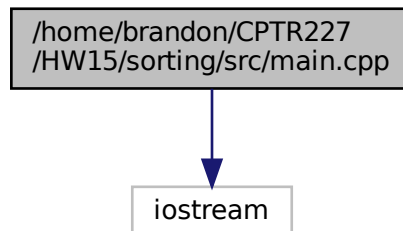
# File Documentation

### 4.1 /home/brandon/CPTR227/HW15/sorting/src/main.cpp File Reference

This is Homework 15-Sorting for CPTR 227.

```
#include <iostream>
```

Include dependency graph for main.cpp:



### Classes

- class `heapsort`

### Functions

- int `main` ()

### 4.1.1 Detailed Description

This is Homework 15-Sorting for CPTR 227.

This is implementation of Heap Sorting.

#### Author

Brandon Yi

#### Date

4/19/2021

### 4.1.2 Function Documentation

#### 4.1.2.1 main()

```
int main ( )
```

Definition at line 97 of file main.cpp.

```
97     {  
98         heapsort h;  
99         cout << "This is before sorting" << endl;  
100        h.print();  
101        h.sort();  
102        cout << "This is after sorting" << endl;  
103        h.print();  
104    }
```

# Index

/home/brandon/CPTR227/HW15/sorting/src/main.cpp, [9](#)

a  
    heapsort, [8](#)

alength  
    heapsort, [8](#)

BinaryHeap  
    heapsort, [5](#)

compare  
    heapsort, [6](#)

heapsort, [5](#)  
    a, [8](#)  
    alength, [8](#)  
    BinaryHeap, [5](#)  
    compare, [6](#)  
    left, [6](#)  
    myswap, [6](#)  
    n, [8](#)  
    parent, [6](#)  
    print, [7](#)  
    reverse, [7](#)  
    right, [7](#)  
    sort, [7](#)  
    trickleDown, [7](#)

left  
    heapsort, [6](#)

main  
    main.cpp, [10](#)

main.cpp  
    main, [10](#)

myswap  
    heapsort, [6](#)

n  
    heapsort, [8](#)

parent  
    heapsort, [6](#)

print  
    heapsort, [7](#)

reverse  
    heapsort, [7](#)

right  
    heapsort, [7](#)

sort  
    heapsort, [7](#)

trickleDown  
    heapsort, [7](#)