●●●● **Not Frequent**                                                                    0/8

# Introduction to Sorting

Authors: Darren Yao, Benjamin Qi, Allen Li, Andrew Wang

*Arranging collections in increasing order.*

Language: Python ⌄                                                      Edit This Page ⧉

**Prerequisites**
- Bronze - Introduction to Data Structures

**TABLE OF CONTENTS**

**Sorting** refers to arranging items in some particular order.

# Sorting Methods

### Bubble Sort ⧉
**HR - Very Easy**                                                                        ⋮

*Focus Problem – try your best to solve this problem before continuing!*

**RESOURCES**

| | | | |
|---|---|---|---|
| CPH | ★ 3.1 - Sorting Theory | bubble sort, merge sort, counting sort | ⋮ |
| CSA | Sorting | selection sort, insertion sort, bubble sort, merge sort | ⋮ |

# Library Sorting

Although you usually do **not** need to know how sorting is implemented, you should know how to use built-in methods.

**RESOURCES**

| | | | |
|---|---|---|---|
| PY | Sorting HOW TO | reference | ⋮ |

## Static Arrays

To create a static array in Python, the `array` module is used. Python does not have a built in sort method for arrays, but you can use Python's `sorted()` function which sorts the array as a list, and returns a list. Then, convert the list back into an array.

```python
from array import array

# "i" denotes integer type of array elements
arr = array("i", [5, 1, 3, 2, 4])

print(arr)  # Outputs the original array
print(sorted(arr))  # Outputs the sorted array, converted to a list

arr = array("i", sorted(arr))  # Sorting, then converting back into an array
print(arr)
```

## Dynamic Arrays

There's two main ways to sort a list in Python. You can use `sorted(arr)`, which returns a new list and doesn't modify the old one, or `arr.sort()`, which sorts the list in place.

```python
arr = [5, 1, 3, 2, 4]
print(sorted(arr))  # Outputs [1, 2, 3, 4, 5]
print(arr)  # Outputs the original array

arr.sort()
print(arr)  # Outputs [1, 2, 3, 4, 5]
```

For more on sorting in Python, see **this link**.

## (Dynamic) Arrays of Pairs & Tuples

By default, Python tuples sort by first element, then second element, and so on in case of repeated ties.

```python
arr = [(1, 5), (2, 3), (1, 2)]
arr = sorted(arr)
print(arr)  # Outputs [(1, 2), (1, 5), (2, 3)]
```

# Problems

⚠ **Warning!**

Bronze problems are designed so that you shouldn't need a $\mathcal{O}(N \log N)$ time sort (repeatedly extracting the minimum in $\mathcal{O}(N^2)$ time will always suffice).