



EXERCISES — Bubble Sort

version #



Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2021-2022 Assistants [<assistants@tickets.assistants.epita.fr>](mailto:assistants@tickets.assistants.epita.fr)

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto some-one else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Bubble Sort	3
1.1	Goal	3
1.2	Prototype	3
1.3	Example	3

*<https://intra.assistants.epita.fr>

1 Bubble Sort

Files to submit:

- bubble_sort/bubble_sort.c

Provided files:

- bubble_sort/bubble_sort.h

Authorized headers: You are only allowed to use the functions defined in the following headers:

- err.h
- errno.h
- assert.h
- stddef.h

1.1 Goal

The bubble sort is a basic sorting algorithm, easy to implement, but quite slow (worst-case and average complexity $\mathcal{O}(n^2)$). It compares each pair of adjacent items and swaps them if they are in the wrong order.

1.2 Prototype

```
void bubble_sort(int array[], size_t size);
```

1.3 Example

```
[6, 1, 8, 5, 4] -> [1, 6, 8, 5, 4]  6 > 1 Swap them  
[1, 6, 8, 5, 4] -> [1, 6, 8, 5, 4]  6 < 8 No need to swap  
[1, 6, 8, 5, 4] -> [1, 6, 5, 8, 4]  8 > 5 Swap them  
[1, 6, 5, 8, 4] -> [1, 6, 5, 4, 8]  8 > 4 Swap them  
...           Start again until the array is fully sorted
```

Your bubble sort will take two arguments: the array to sort, and its size.

It is my job to make sure you do yours.