Bubble Sort

Repeatedly compares adjacent elements from left to right, swapping them if needed, and bubbles the largest unsorted element to the end of the list with each pass.

```python
def bubble_sort(arr: list) -> list:
    end = len(arr)
    sorting = True
    while sorting:
        sorting = False
        for i in range(1, end):
            if arr[i] < arr[i - 1]:
                arr[i], arr[i - 1] = arr[i - 1], arr[i]
                sorting = True
        end -= 1
    return arr
```

Property	Details
How it works	Basically, it moves the highest number to the end of the list on each iteration, then reduces the range for the next iteration. The algorithm compares adjacent elements from left to right. If they are in the wrong order, they are swapped. After each pass, the largest unsorted element bubbles up to its correct position at the end of the list. This process repeats until the entire list is sorted.
Time Complexity	$O(n^2)$ in the worst and average case, $O(n)$ in the best case (already sorted)
Space Complexity	$O(1)$
Stable	Yes
Type	in-place, iterative

- Follow on Bluesky → https://bsky.app/profile/leonlonsdale.dev
- Join Discord → https://discord.gg/dhrdFh98UA