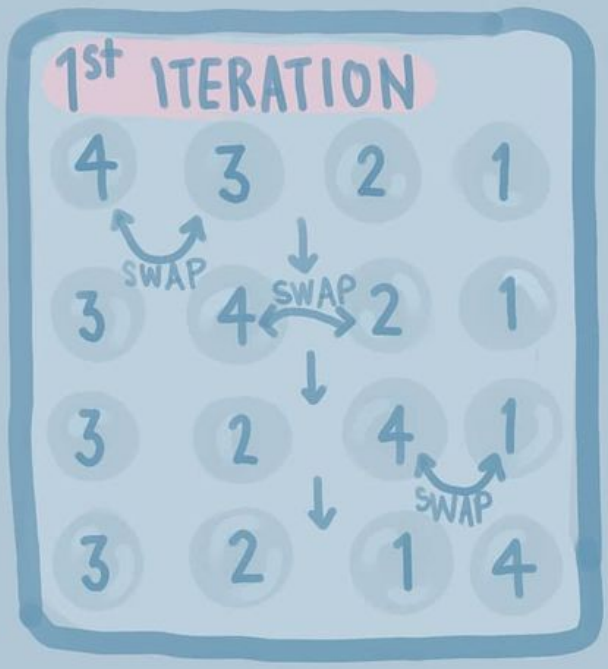
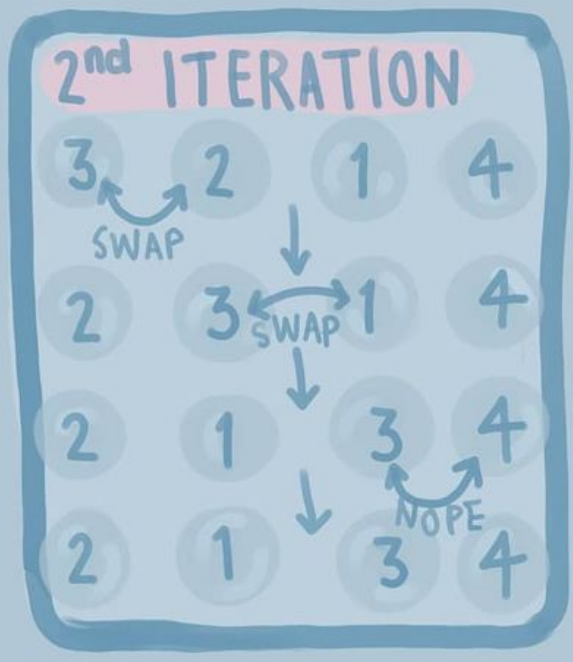
BUBBLE SORT

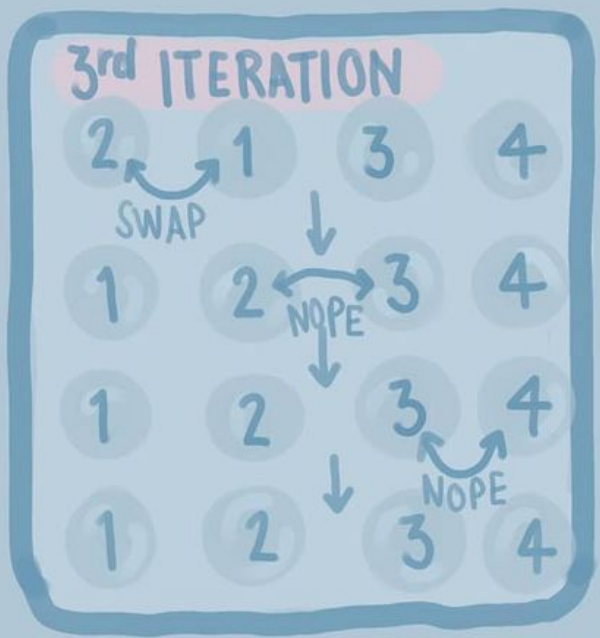
- Start from the beginning of the array and swap the adjacent elements if the right element is greater than the left (i.e. increasing order)
- Move onto the next element and repeat till you reach the end of the array
- By the end, the greatest element would have `bubbled down' to the right end
- Repeat another iteration for the second greatest element
- The right end of the array is sorted
- Atmost `n' (length of array) iterations are needed to completely sort the array.

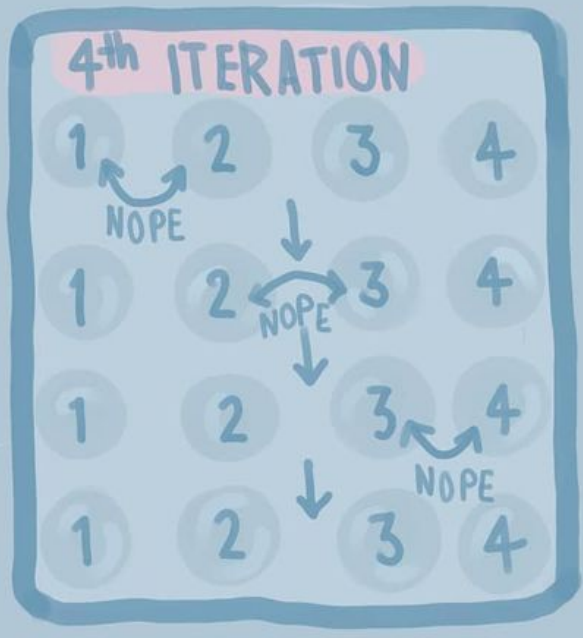
BUBBLE SORT

LET'S SEE HOW IT WORKS!









BUBBLE SORT

TIME COMPLEXITY

BEST

n → already sorted

AVERAGE

n^2

WORST

n^2 → reverse order

SPACE COMPLEXITY

1

- The algorithm stops when there are no swaps in one complete iteration
- We need to perform atleast one iteration after the array is sorted
- In-place sorting