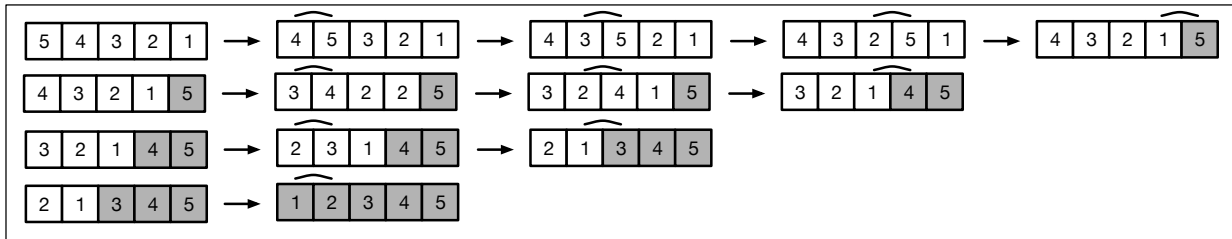


Algorithmic Patterns 1: Solution



teachinglondoncomputing.org

10

The algorithm here is Bubblesort.

Each pass compares adjacent pairs in turn moving down the array. The pass takes the largest number found so far towards the end of the array. At the end of the pass that largest number is deposited at the end in the right place. That number does not need to be checked on the next pass so the passes get shorter, but each pass still deposits another number in the right place.

The last pass just does one comparison, putting another number in the right place (the second smallest number). However when that is in the right place, the last number has nowhere else to go so it is automatically in the right place. This means the number of passes is one less than the length of the array.

As you make the array larger and add one more number to the array, so increasing the length of the array by one, you have to do one more pass. That pass involves one less comparison than the length of the array, so the total number of comparisons needed goes up by that amount.