

Programming Club Meeting 15

Slides

—

Bubble and Selection Sort

Quick Note

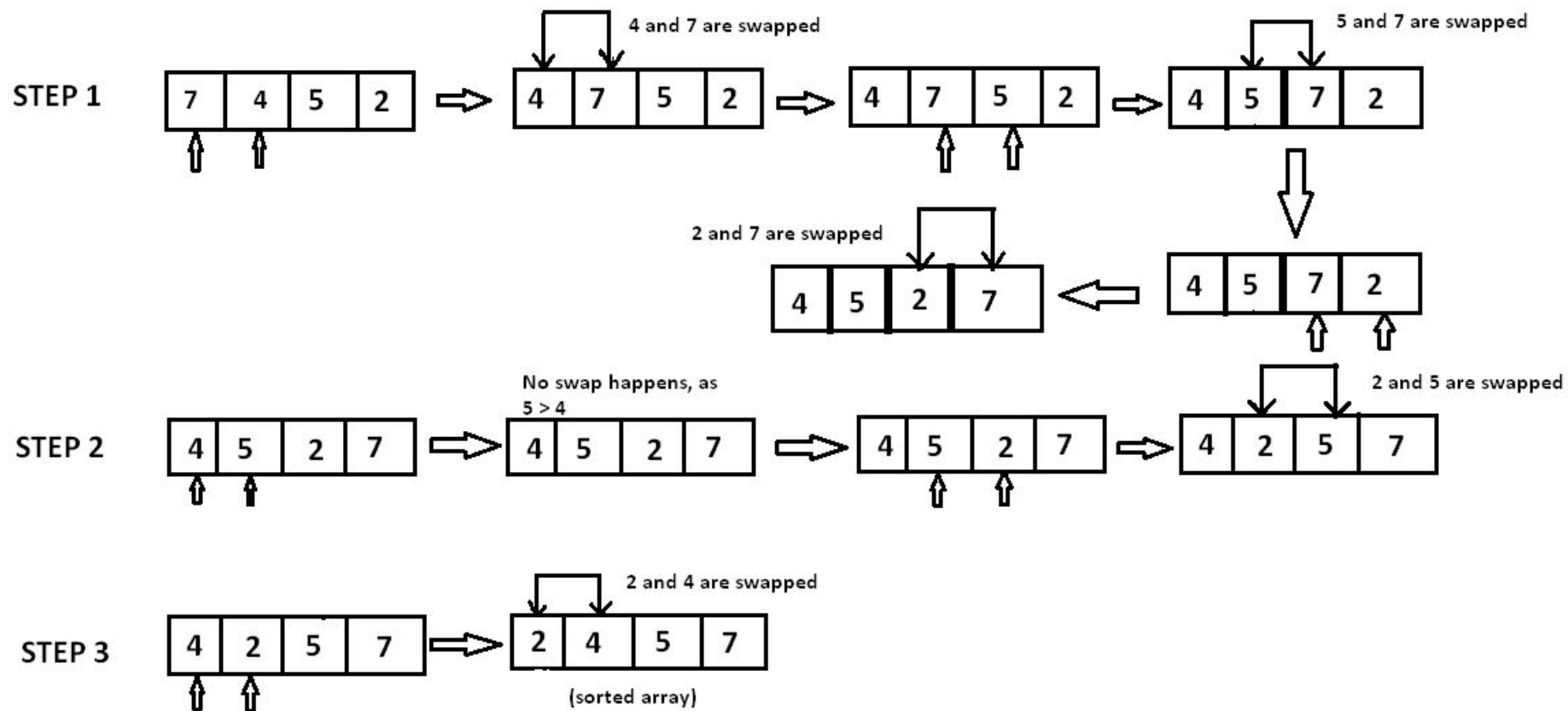
- Python's built in sort method uses Tim Sort which is way quicker than anything you or I could write for most things, but learning how it works is helpful
- All sorting algorithms will work for all lists, but some are better than others depending on what exactly needs to be done

Bubble Sort



Bubble Sort

- Check the 1st and 2nd element, if they are out of order, swap them
 - Do this for the 2nd and 3rd element
 - Keep moving 1 forward and swapping out of place elements until you reach the end of the list
 - Do this until the list is sorted
 - Very easy method to implement, but fairly slow with long lists
-



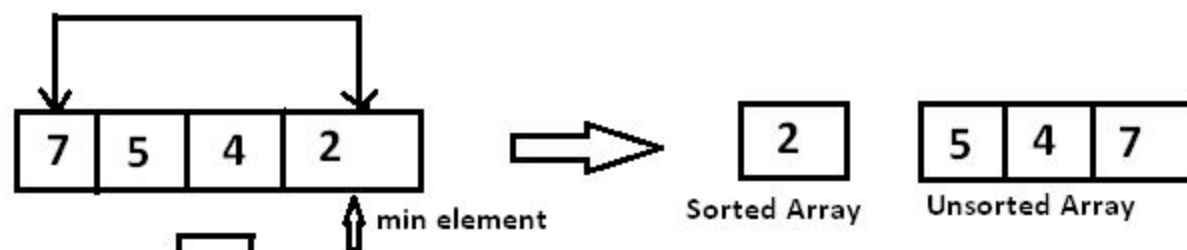
Selection Sort



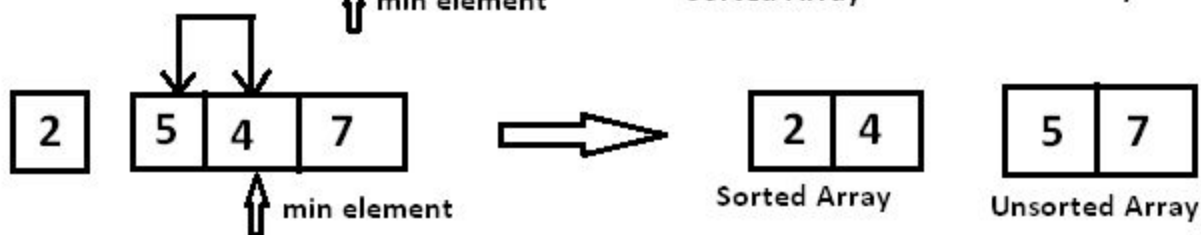
Selection Sort

- Go through each unsorted element in the list and find the one with the lowest value
- Swap this element with the first unsorted value (meaning that it is now sorted)
- Repeat this until the list is sorted
- Also quite easy to implement, still fairly slow with large lists
- Can also technically find the highest element and push it to the end, but typically the lowest is used

STEP 1.



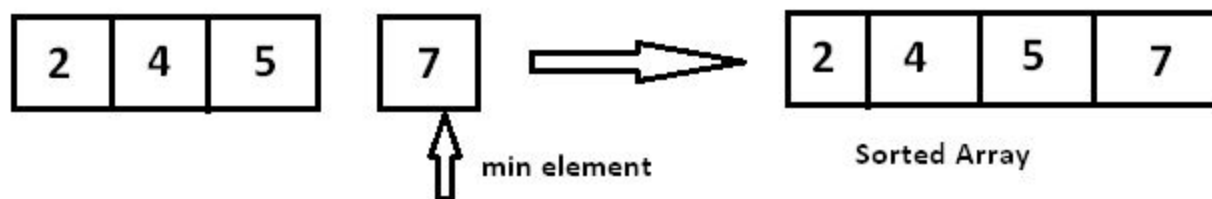
STEP 2.



STEP 3.



STEP 4.



Practice Problems

—

Practice Problem 1:

Implement Bubble Sort

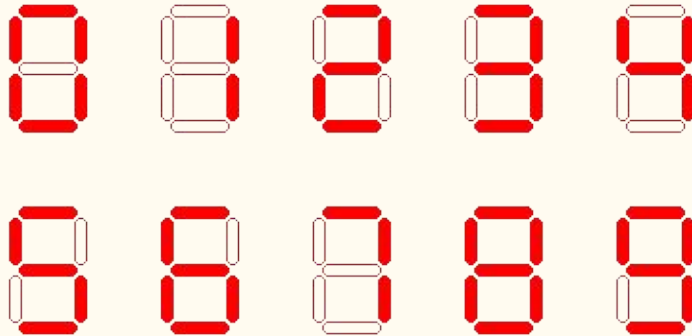
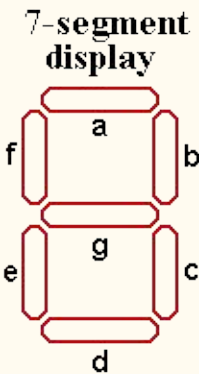
- Goal: Write a Python program that implements a bubble sort function.
 - Relevant Information:
 - Ex Input: [5, 1, 4, 2, 8]
 - Want to go through the list swapping out of place elements until the list is sorted.
-

Practice Problem 2:

- Goal: Write a Python program that implements a selection sort function.
- Relevant Information:
 - Ex Input: [64, 25, 12, 22, 11]
 - Want to go through the list swapping the lowest unsorted element with the first unsorted element until the list is sorted.

Practice Problem 3:

Matchstick Display



- Src:
<https://www.hackerearth.com/practice/basic-programming/input-output/basics-of-input-output/practice-problems/algorithm/seven-segment-display-nov-easy-e7f87ce0/>
- Goal: Write a Python program that will output the largest value that can be created on a 7-segment display when each lit segment is represented by a matchstick. Note that the total number of matchsticks that can be used is limited and to be inputted by the user.
- Relevant Information:
 - Ex 1: Input: 2, Output: 1
 - Ex 2: Input 6, Output: 111
 - Ex 3: Input 11, Output: 71111

Next Meeting: Big O Notation

