Bubble Sort

Authored by AllisonP

Table Of Contents

Bubble Sort

This is a very simple sorting algorithm. Because it's also very inefficient, Bubble Sort is not practical for real-world use and is generally only discussed in an academic context. The basic theory behind BubbleSort is that you take an array of integers and iterate through it; for each element at some index i whose value is *greater than* the element at the index following it (i.e., index $i + 1$), you must swap the two values. The act of swapping these values causes the larger, unsorted values to float to the back (like a bubble) of the data structure until they land in the correct location.

Asymptotic Analysis

- Worst Case: $\mathcal{O}(n^2)$
- Best Case: $\mathcal{O}(n)$
- Average: $\mathcal{O}(n^2)$

Example (Java)

\- EXAMPLE

Take some time to review the Java implementation of BubbleSort below.

Input Format
The first line contains an integer, n, denoting the number of elements to be sorted.
The second line contains n space-separated integers denoting the respective elements of the unsorted array.

```
1  import java.util.*;
2
3  class Sorting {
4      private static void printArray(String s, int[] x) {
5          System.out.print(s + " Array: ");
6          for(int i : x){
7              System.out.print(i + " ");
8          }
9          System.out.println();
10     }
11
12     public static void bubbleSort(int[] x) {
13         printArray("Initial", x);
14
15         int endPosition = x.length - 1;
16         int swapPosition;
17
18         while( endPosition > 0 ) {
19             swapPosition = 0;
20
21             for(int i = 0; i < endPosition; i++) {
22
23                 if( x[i] > x[i + 1] ){
24                     // Swap elements 'i' and 'i + 1':
25                     int tmp = x[i];
26                     x[i] = x[i + 1];
27                     x[i + 1] = tmp;
28
29                     swapPosition = i;
30                 } // end if
31
32                 printArray("Current", x);
33             } // end for
34
35             endPosition = swapPosition;
36         } // end while
37
38         printArray("Sorted", x);
39     } // end bubbleSort
40
41     public static void main(String[] args) {
42         Scanner scanner = new Scanner(System.in);
43         int n = scanner.nextInt();
44         int[] unsorted = new int[n];
45         for(int i = 0; i < n; i++) {
46             unsorted[i] = scanner.nextInt();
47         }
48         scanner.close();
49
50         bubbleSort(unsorted);
51     }
52 }
```

Input

```
5
1 5 2 4 3
```

Run

Output