Data Structures - Simple Sorting 2

Dr. TGI Fernando [1] [2]

[1] Email: tgi.fernando@gmail.com
[2] URL: http://tgifernando.wordpress.com/

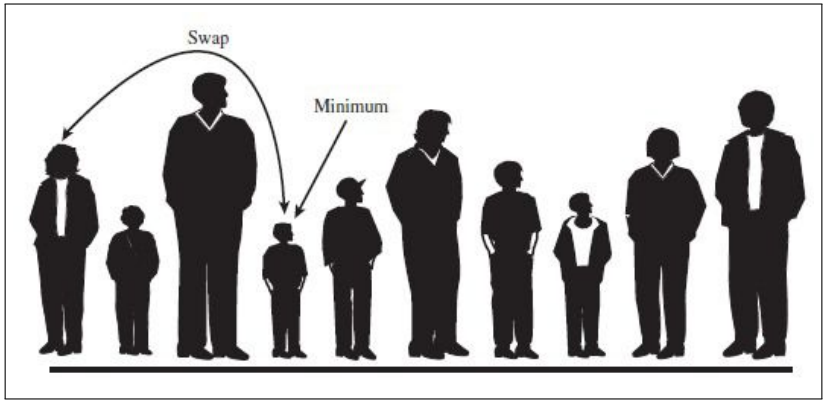
Description

- In the first pass, algorithm picks the shortest player (or **selecting**, hence the name of the sort).
- This shortest player is then swapped with the player on the left end of the line, at position 0.
- Now the leftmost player is sorted and won't need to be moved again.
- Sorted players are on the left (lower indices), whereas in the bubble sort they are on the right.
- In the second pass, algorithm starts at position 1, and, finding the minimum, swap with position 1.
- This process continues until all the players are sorted.

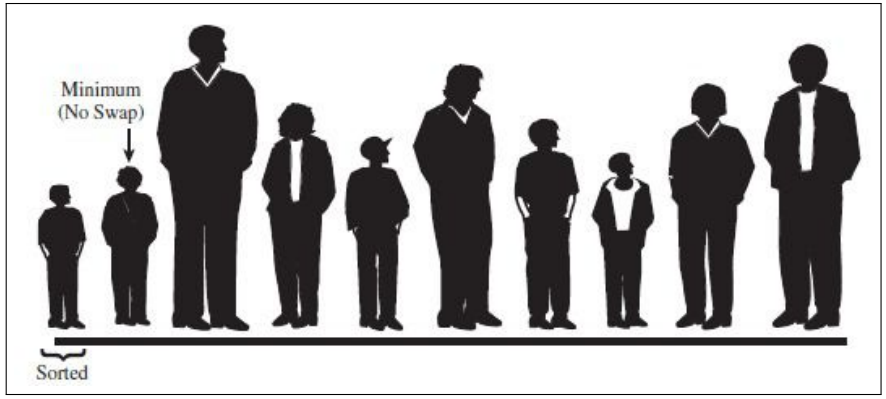
Selection Sort

- Improves the bubble sort by reducing the no of swaps from $O(N^2)$ to $O(N)$.
- However, the no of comparisons remains $O(N^2)$.
- Offers a significant improvement when large records that must be physically swapped.
- But this is not a case in Java, where references are moved around, not entire objects.

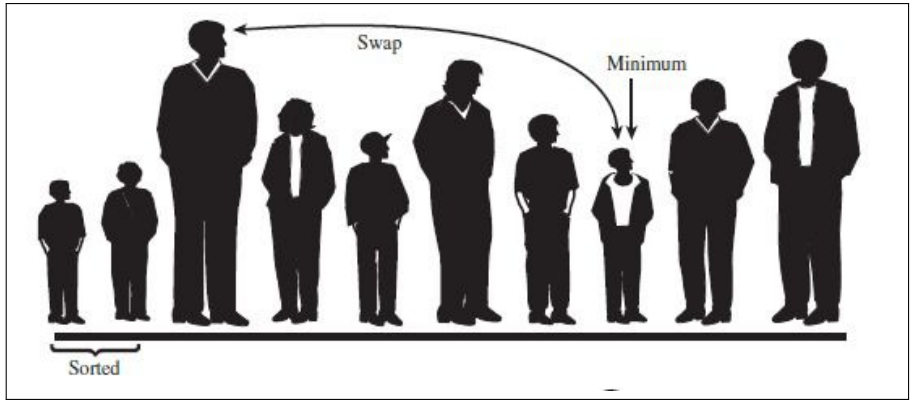
First pass



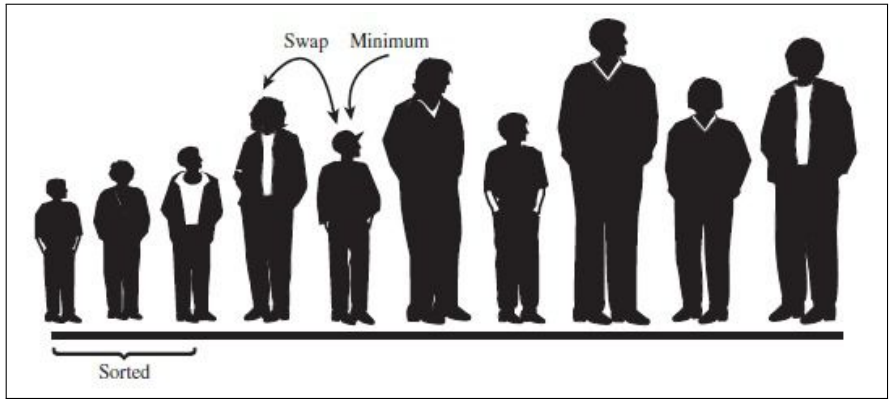
Second pass



Third pass



Fourth pass



Efficiency of the Selection Sort

- Performs the same no of comparisons as the bubble sort: $N(N-1)/2$.
- No of swaps $\leq N$.
- Selection sort runs in $O(N^2)$ time as the bubble sort did.
- However, the selection sort is faster because there are few swaps.