

STALIN'S SORT: SORTING BY REMOVAL

Ivanchikov M. M., student

Antsuto A. Y., student

Scientific supervisor – Vanik I. Y., senior lecturer

English language department №1

Belarusian National University of Technology

Minsk, Republic of Belarus

Nowadays, sorting algorithms play a vital role in computer science, since they can reduce the complexity of a problem. Sorting algorithms consist of a set of instructions that take an array as an input and put the items in a special order. Among these algorithms, Stalin's Sort stands out as a non-linear but thoughtful approach.

Stalin's sort works on a radical principle: instead of rearranging the elements, it destroys the disordered ones. The result of this process is a well-sorted list – but only by removing all disordered items. The algorithm keeps the first item and keeps the next item if they maintain a non-decreasing order compared to the previous correct item. Everything else is gone [1].

The sorting algorithm traditionally aims to rearrange the elements to obtain an order with the least amount of change. Stalin's sort challenges this view by adopting a destructive but effective strategy. Its simplicity makes it a good example of extreme design philosophies in algorithm development.

The pseudo code of Stalin's sort can be expressed as follows:

FUNCTION stalinSort(A : list OF sortable items)

 n := length(A)

 bigger := 0

 B SET empty list

 FOR i := 0 TO n NOT inclusive DO

 IF A[i] >= bigger THEN

 bigger := A[i]

 B.push(A[i])

 END IF

 END FOR

RETURN B

END FUNCTION [2]

The performance of an algorithm is highly data dependent. In the best case when the input list is sorted—the time complexity is $O(n)$. In the worst case when the list shrinks dramatically – the result contains only the first element, which still runs in $O(n)$ time. The complexity of the field is also $O(n)$ because it stores the largest valid subsequence.

It's worth noting that Stalin's Sort prioritizes order over completeness. The algorithm does not attempt to restore data or balance accuracy and efficiency. This approach highlights an important aspect of algorithm design: a trade-off between accuracy, efficiency, and problem constraints.

In the early 2020s, meme culture and computer science enthusiasts popularized Stalin's piece. These extreme strategies became a funny example of how technically the problem could be solved. However, no one was using them in practice. Stalin's sort is instructive, showing how attitudes influence the final approach to problem solving. This kind of algorithm is in the group of the most bizarre sorting algorithms, including Thanos Sort, Sleepy Sort, Quantum BogoSort and others. They cheerfully illustrate that even flawed or unwieldy algorithms can offer a delightful experience [3].

Finally, Stalin's sort is about exploring the limits of designing algorithms rather than a practical one. It serves as a reminder that while there are many solutions to a given problem, not all are effective – but that even the ones that don't work can spark discussion and a deeper understanding.

References

1. The ultimate sorting algorithm—URL: <https://medium.com/@kaweendra/the-ultimate-sorting-algorithm-6513d6968420> (date of access: 25.03.2025).

2. Stalin sort. – URL: <https://github.com/gustavo-depaula/stalin-sort> (date of access: 25.03.2025).

3. 7 Weird and whacky sorting algorithms you've probably never heard of – URL: <https://www.linkedin.com/pulse/7-weird-whacky-sorting-algorithms-youve-probably-never-chinnaraj-acare/> (date of access: 25.03.2025).