Selection Sort

Selection Sort

Selection Sort is a `simple comparison-based` sorting algorithm.

It divides the array into `two parts` : a **sorted subarray** and an **unsorted subarray**.

Initially, the **sorted part is empty**, and the unsorted part is the entire array.

In `each iteration` , it finds the minimum element from the **unsorted part** and moves it to the end of the `sorted part` .

Example 1:

Input: [4, 5, 1, 3, 9]

Output: [1, 3, 4, 5, 9]

Approach:

Iterate over the array from index `0` to `n-2` .

For each index `i` , assume the element at `i` is the `minimum` in the **unsorted part**.

Run an inner loop from `j = i+1 to n-1` to find the actual minimum element.

If a smaller element is found, update the `min index` .

After the **inner loop, swap the element** at `i` with the element at min (if they're not the same).

`Repeat` until the **array is sorted**.

Time & Space Complexity:

Time Complexity: O(n^2) In all cases `best, average and worst` .

Roughly `n*(n-1)/2` comparisons are always performed.

Space Complexity: O(1) Selection Sort is an in-place sorting algorithm, so it doesn't require extra space.

Dry Run

Input: `arr = [4, 5, 1, 3, 9]`

`i = 0 → min_idx = 0 j = 1 → arr[1] = 5 > arr[0] = 4 → no change j = 2 → arr[2] = 1 < arr[0] = 4 → min_idx = 2 j = 3 → arr[3] = 3 > arr[2] = 1 → no change j = 4 → arr[4] = 9 > arr[2] = 1 → no change swap arr[0] ↔ arr[2] → [1, 5, 4, 3, 9] i = 1 → min_idx = 1 j = 2 → arr[2] = 4 < arr[1] = 5 → min_idx = 2 j = 3 → arr[3] = 3 < arr[2] = 4 → min_idx = 3 j = 4 → arr[4] = 9 > arr[3] = 3 → no change swap arr[1] ↔ arr[3] → [1, 3, 4, 5, 9] i = 2 → min_idx = 2 j = 3 → arr[3] = 5 > arr[2] = 4 → no change j = 4 → arr[4] = 9 > arr[2] = 4 → no change no swap → [1, 3, 4, 5, 9] i = 3 → min_idx = 3 j = 4 → arr[4] = 9 > arr[3] = 5 → no change no swap → [1, 3, 4, 5, 9]` **Final Sorted Array:** `[1, 3, 4, 5, 9]`

JavaScript | Python | Java | C++ | C | C#

```
let arr = [4, 5, 1, 3, 9];

function selectionSort(arr) {
  let n = arr.length;
  for (let i = 0; i < n - 1; i++) {
    let min = i;
    for (let j = i + 1; j < n; j++) {
      if (arr[j] < arr[min]) {
        min = j;
      }
    }
    if (min != i) {
      let temp = arr[i];
      arr[i] = arr[min];
      arr[min] = temp;
```

```
        }
      }
      return arr;
    }

    let result = selectionSort(arr);
    console.log("Sorted array", result);
```


Selection Sort - DSA Notes

Selection Sort - DSA Notes