Insertion and Selection Sort

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Atik Mohtasin Rahi
Roll: 2003118
Department: CSE

Barkatulla Barik
Roll: 2003071
Department: CSE

Mir Ashikur Rahman
Roll: 2003109
Department: CSE

Rajshahi University of Engineering & Technology

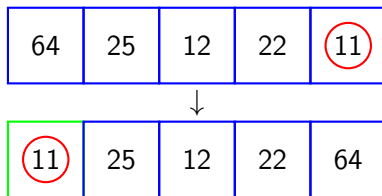January 20, 2024

Welcome

Welcome to our Presentation!

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Outline

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

First Topic

Selection Sort

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Selection Sort: First Pass

- For the first position in the sorted array, traverse the entire array.
- Find the minimum value (11) and swap it with the element at the first position (64).

64	25	12	22	(11)

$\downarrow$

(11)	25	12	22	64

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Selection Sort: Second Pass

- For the second position, find the second minimum value (12) and swap it with the element at the second position (25).

| 11 | 25 | (12) | 22 | 64 |

↓

| 11 | (12) | 25 | 22 | 64 |

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE
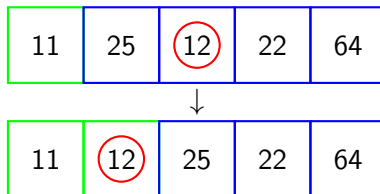
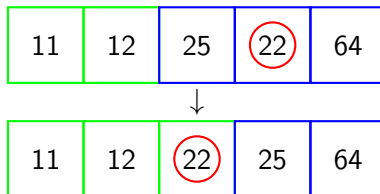# Selection Sort: Third Pass

- For the third position, find the third minimum value (22) and swap it with the element at the third position (25).

| 11 | 12 | 25 | (22) | 64 |

↓

| 11 | 12 | (22) | 25 | 64 |

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Selection Sort: Fourth Pass

- For the fourth position, find the fourth minimum value (25) and never swap it.

11	12	22	(25)	64

↓

11	12	22	(25)	64

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE
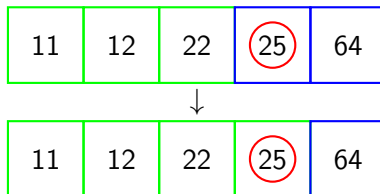
Selection Sort: Fifth Pass

- The largest value (64) is automatically placed at the last position.

11	12	22	25	(64)

$\downarrow$

11	12	22	25	(64)

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE
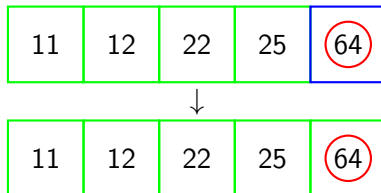
Selection Sort Algorithm: Overview

Algorithm 1: Selection Sort

Data: Array *arr* of size *n*
Result: Sorted array *arr*

1 **for** $i \leftarrow 0$ **to** $n - 1$ **do**
2      $min\_idx \leftarrow i$;
3   **for** $j \leftarrow i + 1$ **to** n **do**
4   **if** $arr[j] < arr[min\_idx]$ **then**
5             $min\_idx \leftarrow j$;

6   **if** $min\_idx \neq i$ **then**
7   Swap($arr[min\_idx]$, $arr[i]$);

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

Selection Sort: C++ Code

```cpp
void selectionSort(int arr[], int n)
{
    int i, j, min_idx;
    for (i = 0; i < n - 1; i++) {
        min_idx = i;
        for (j = i + 1; j < n; j++) {
            if (arr[j] < arr[min_idx])
                min_idx = j;
        }
        if (min_idx != i)
            swap(arr[min_idx], arr[i]);
    }
}
```

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Second Topic

Insertion Sort

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Insertion Sort: Step-by-Step

- For each element, insert it into its correct position in the sorted portion of the array.
- Shift elements greater than the key to the right.
- Repeat until the entire array is sorted.

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
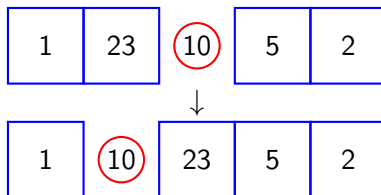Department: CSE   Department: CSE   Department: CSE

Insertion Sort: First Pass

- Initially, the first two elements of the array are compared in insertion sort.
- Here, 23 is greater than 1 hence they are not in the ascending order and 23 is not at its correct position. Thus, swap 1 and 23.
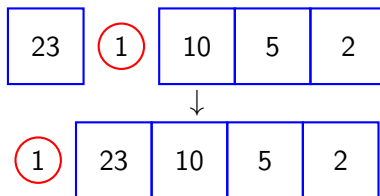
Insertion Sort: Second Pass

- Here, 23 is greater than 10 hence they are not in the ascending order and 10 is not at its correct position. Thus, swap 1 and 23. 10 also stored in a sorted sub-array along with 1
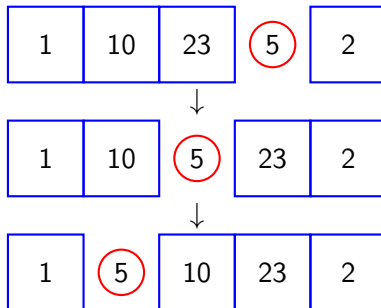
| 1 | 23 | (10) | 5 | 2 |

↓

| 1 | (10) | 23 | 5 | 2 |

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

Insertion Sort: Third Pass

- Here, 5 isn't in correct position. So 5 has to be swapped with its previous position until 5 isn't greater than the previous value.

1	10	23	(5)	2

↓

1	10	(5)	23	2

↓

1	(5)	10	23	2

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Insertion Sort: Forth Pass

- Here, 2 isn't in correct position. To place 2 in correct position, we have to follow the same procedure as third pass.

| 1 | 5 | 10 | 23 | (2) |

↓

| 1 | 5 | 10 | (2) | 23 |

↓

| 1 | 5 | (2) | 10 | 23 |

↓

| 1 | (2) | 5 | 10 | 23 |

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Param Paul
Roll: 2003078
Department: CSE
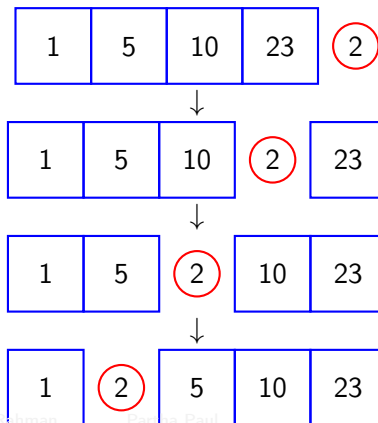
Insertion Sort: Visualization

Figure: Insertion Sort Visualization

Md. Abu Sufyan   Tameem Rahman   Partha Paul
   Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE
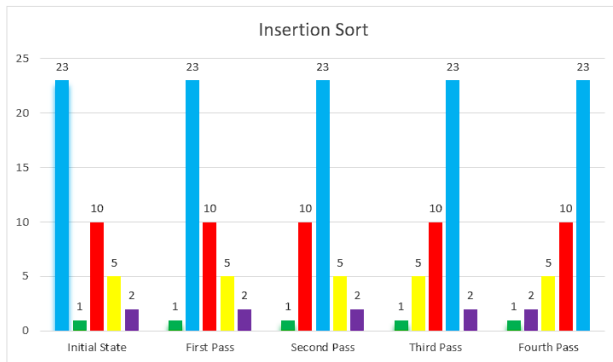
Insertion Sort Algorithm: Overview

Algorithm 2: Insertion Sort

Data: Array *arr* of size *n*

Result: Sorted array *arr*

1 **for** $i \leftarrow 1$ **to** *n* **do**
2 $\quad$ $key \leftarrow arr[i]$;
3 $\quad$ $j \leftarrow i - 1$;
4 $\quad$ **while** $j \geq 0$ **and** $arr[j] > key$ **do**
5 $\quad\quad$ $arr[j + 1] \leftarrow arr[j]$;
6 $\quad\quad$ $j \leftarrow j - 1$;
7 $\quad$ $arr[j + 1] \leftarrow key$;

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

Insertion Sort: C++ Code

```cpp
void insertionSort(int arr[], int n) {
    for (int i = 1; i < n; i++) {
        int key = arr[i];
        int j = i - 1;

        while (j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j = j - 1;
        }

        arr[j + 1] = key;
    }
}
```

Md. Abu Sufyan   Tasmeem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

Conclusion

- Insertion Sort is a simple and intuitive sorting algorithm.
- It efficiently builds the final sorted array one element at a time.
- While not as efficient on large datasets as more advanced algorithms, it performs well on small datasets or nearly sorted datasets.

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

References

- Author et al., *Introduction to Algorithms*, 3rd Edition.
- Author and Collaborator, *Journal of Sorting Algorithms*, 20XX.

Md. Abu Sufyan   Tameem Rahman   Partha Paul
Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

References for Insertion Sort

- Author et al., *Journal of Sorting Algorithms*, 20XX. https://example.com/paper1
- Author and Collaborator, *Conference on Algorithms*, 20XX. https://example.com/paper2

Md. Abu Sufyan   Tameem Rahman   Partha Paul
  Roll: 2003085   Roll: 2003089   Roll: 2003078
Department: CSE   Department: CSE   Department: CSE

References for Selection Sort

- Author et al., *Journal of Sorting Algorithms*, 20XX. https://example.com/paper1
- Author and Collaborator, *Conference on Algorithms*, 20XX. https://example.com/paper2

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Questions & Answers

Any Questions?

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE

Thanks

Thank You for Your Attention!

Contact: your.email@example.com

Md. Abu Sufyan
Roll: 2003085
Department: CSE

Tameem Rahman
Roll: 2003089
Department: CSE

Partha Paul
Roll: 2003078
Department: CSE