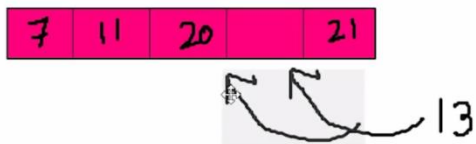
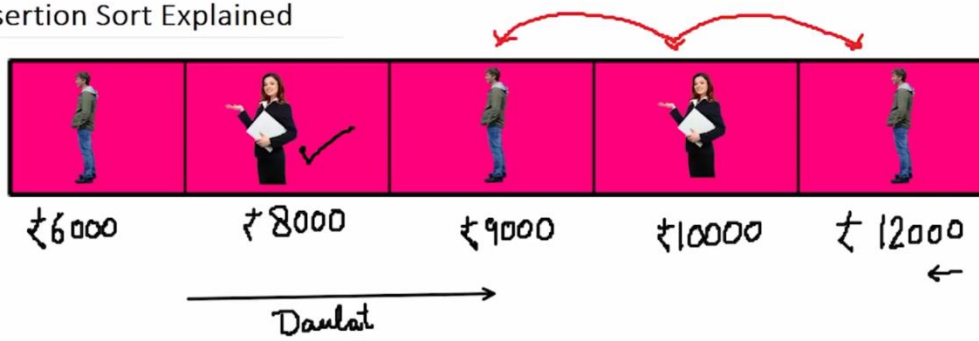
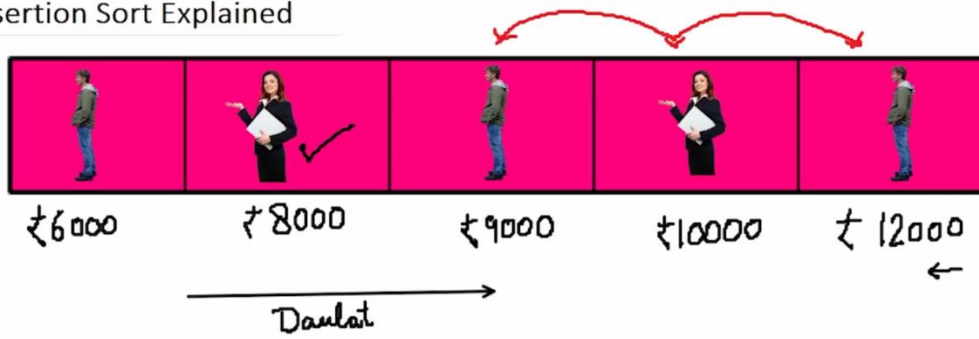


Insertion Sort Explained



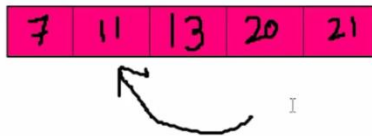
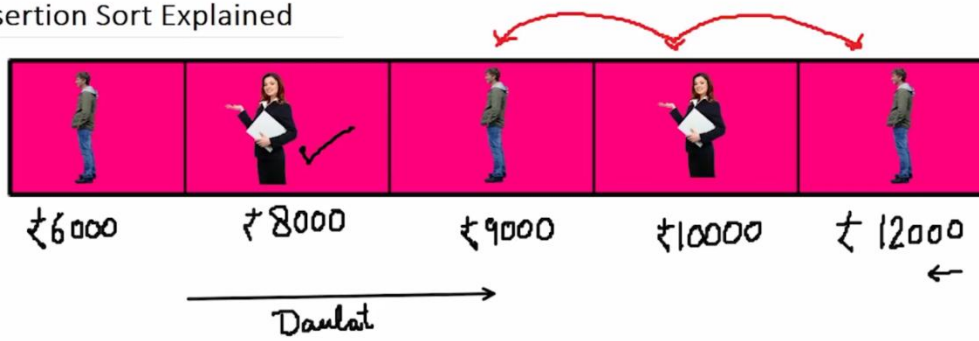
Lalita, ✓

Insertion Sort Explained



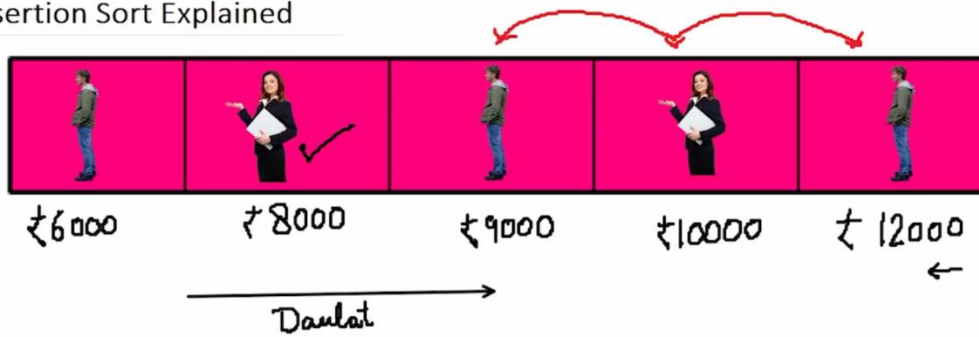
Lalita, ✓

Insertion Sort Explained



Lalita: ✓

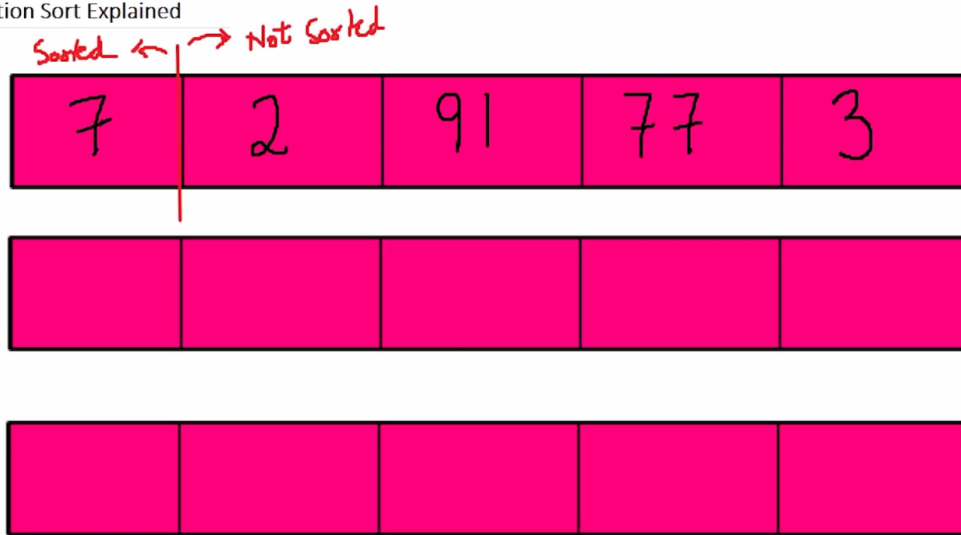
Insertion Sort Explained



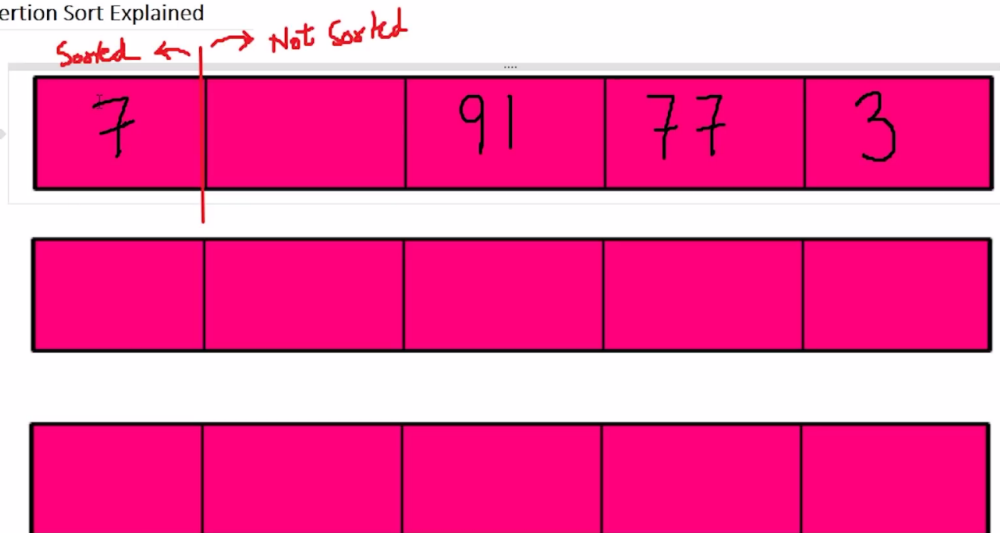
Lalita: ✓

Inserting an element in a Sorted Array.

Insertion Sort Explained

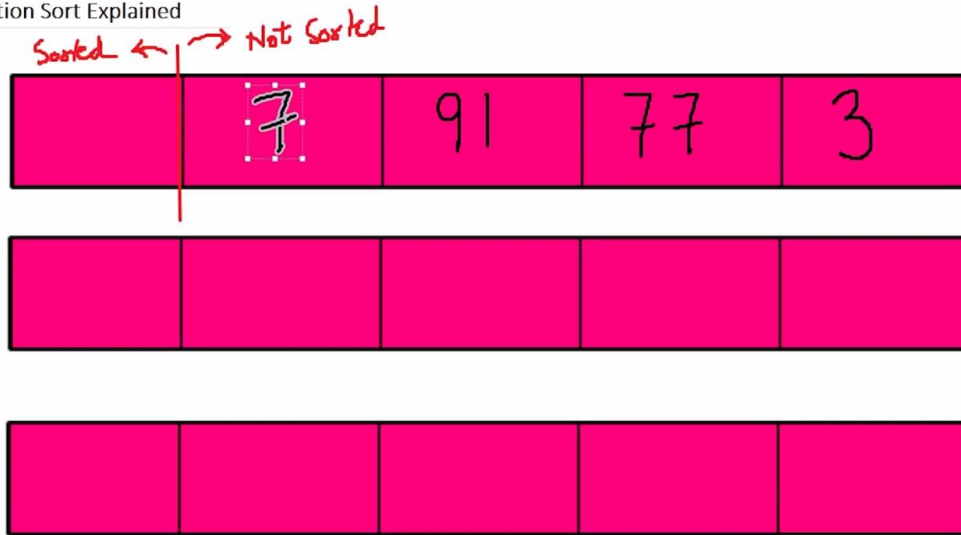


Insertion Sort Explained



2

Insertion Sort Explained



I

2

Insertion Sort Explained



Insertion Sort Explained

S. ← → NS

2	7	91	77	3
---	---	----	----	---

--	--	--	--	--

--	--	--	--	--

Insertion Sort Explained

91

S. ← → NS

2	7		77	3
---	---	--	----	---

--	--	--	--	--

--	--	--	--	--

9

Insertion Sort Explained

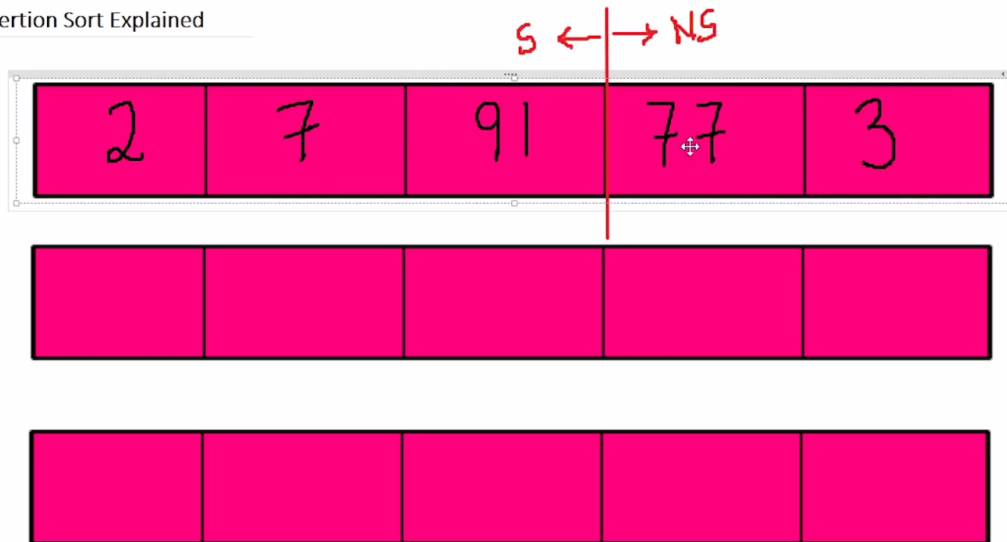


Diagram illustrating the Insertion Sort algorithm. The array contains the numbers 2, 7, 91, and 3. A vertical line is drawn between 91 and 3, with the label "S ← → NS" above it, indicating the current element being inserted (3) and the position where it should be inserted (between 91 and 3). To the right of the array, the number 77 is written, indicating the element that was previously at the position of 3. Below the array are two empty rows of five cells each, representing the state of the array during the sorting process.

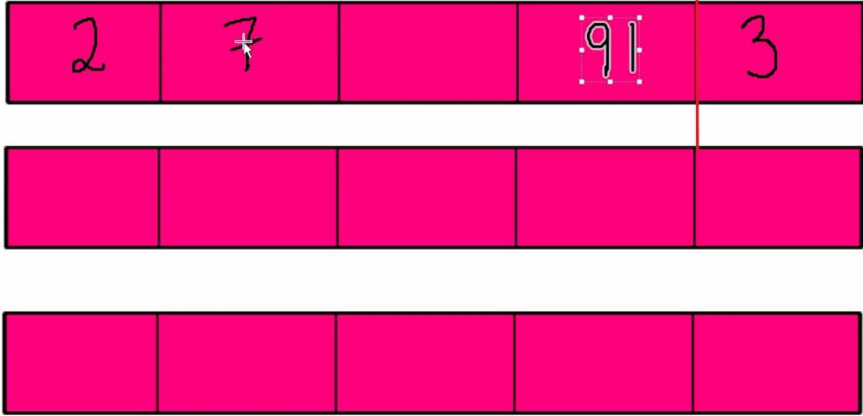
52 new Insertion Sort Algorithm in Hindi.mp4 - VLC media player

Media Playback Audio Video Subtitle Tools View Help

11:50 45% ~20:26

52 new Insertion Sort Algorithm in Hindi.mp4 - VLC media player
Media Playback Audio Video Subtitle Tools View Help

Insertion Sort Explained

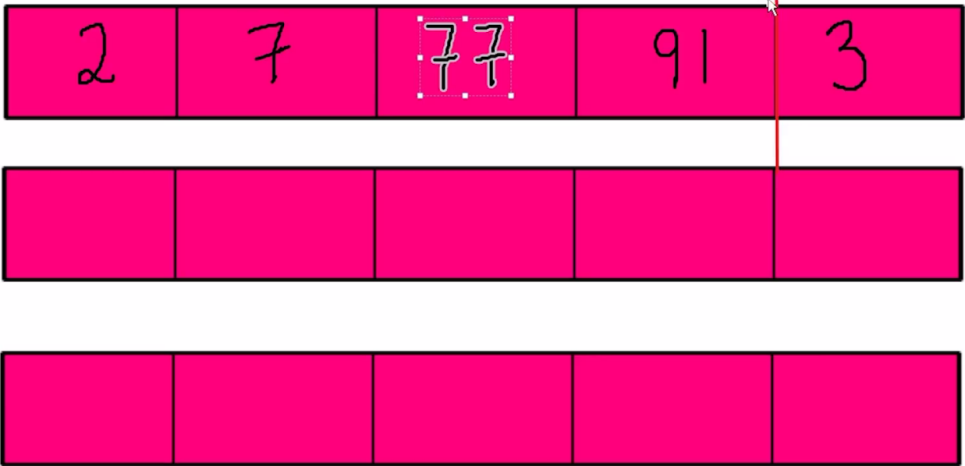


The diagram illustrates the first step of the Insertion Sort algorithm. It features three horizontal rows of five magenta boxes each. The top row contains the numbers 2, 7, an empty box, 91, and 3. The number 7 is highlighted with a dashed border, and a red arrow points from it to the empty box. Above the top row, a red vertical line is drawn between the third and fourth boxes, with 'S' to its left and 'NS' to its right, and a double-headed red arrow above it. To the right of the middle row, the number '77' is written. The bottom row is empty.

77

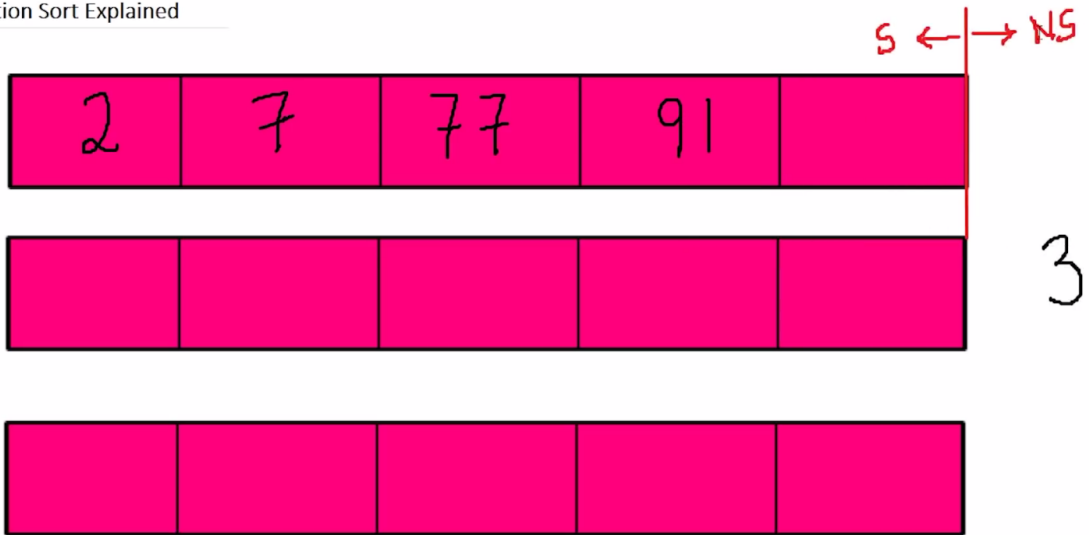
19:12 7:22 PM 23-Apr-22

Insertion Sort Explained

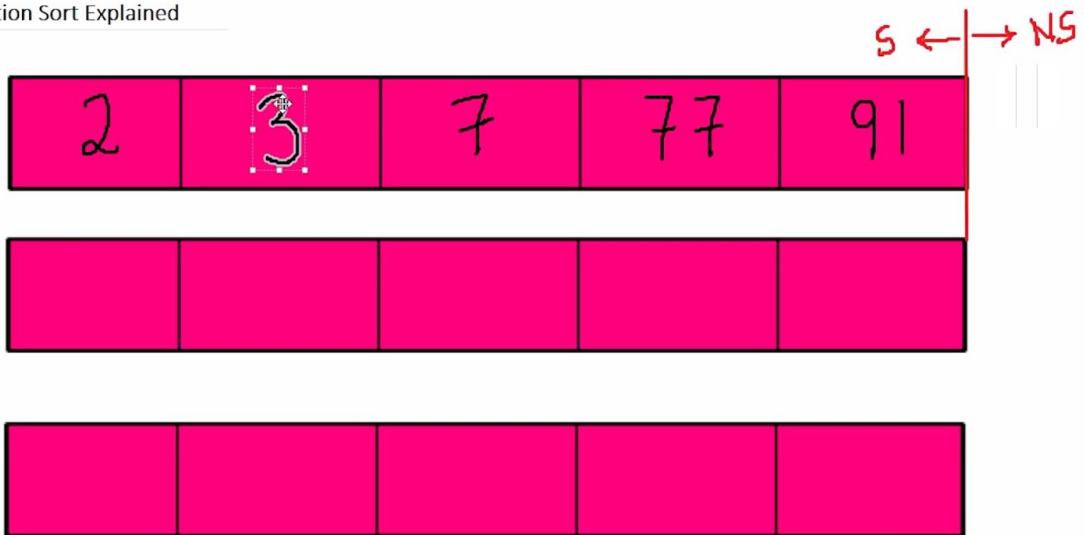


The diagram illustrates the second step of the Insertion Sort algorithm. It features three horizontal rows of five magenta boxes each. The top row contains the numbers 2, 7, 77, 91, and 3. The number 77 is highlighted with a dashed border, and a red arrow points from it to the empty box. Above the top row, a red vertical line is drawn between the fourth and fifth boxes, with 'S' to its left and 'NS' to its right, and a double-headed red arrow above it. The middle and bottom rows are empty.

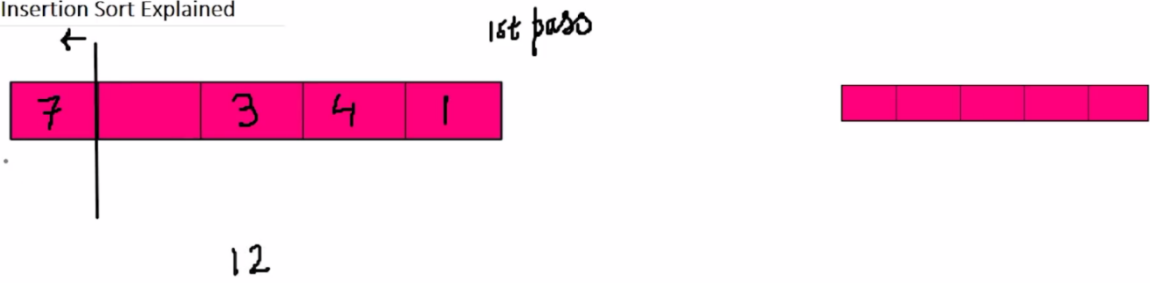
Insertion Sort Explained



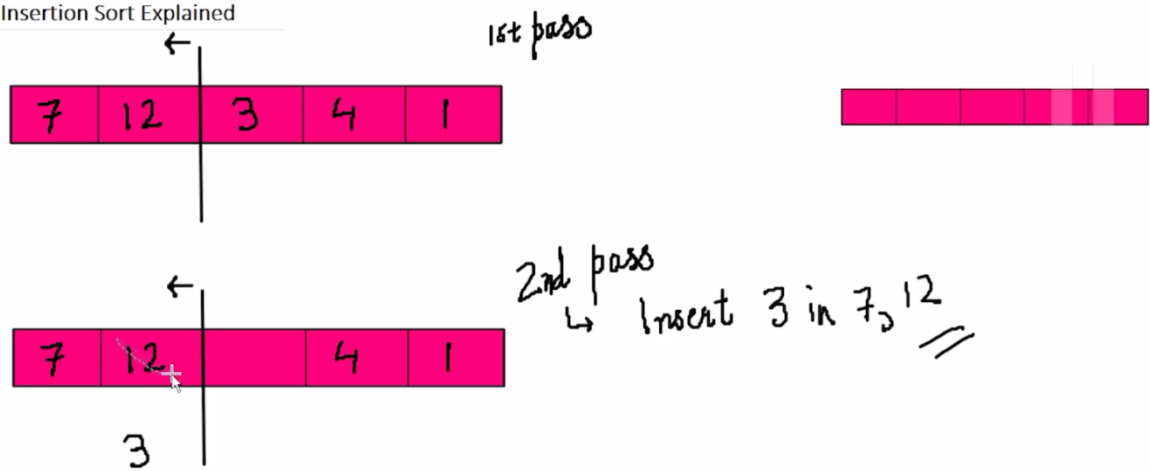
Insertion Sort Explained



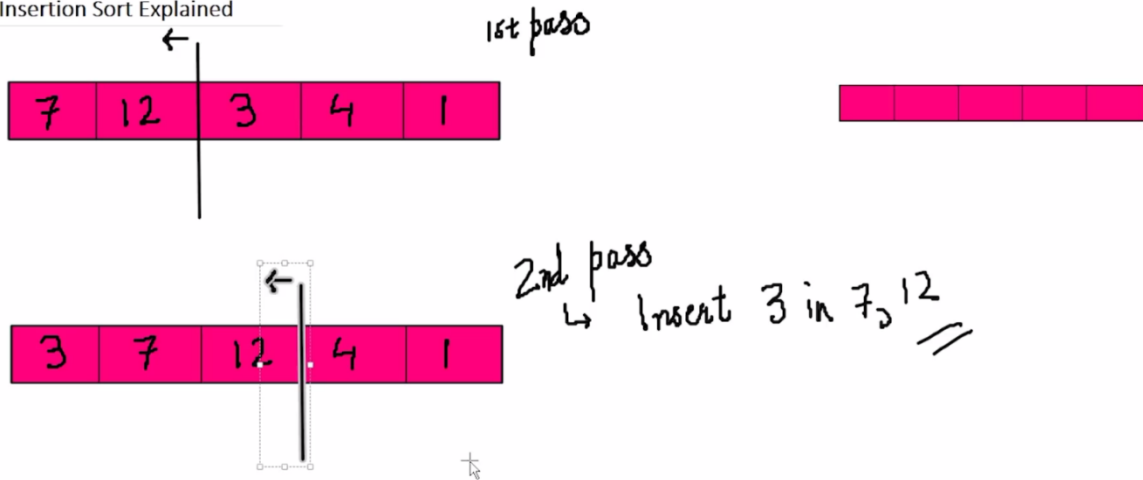
Insertion Sort Explained



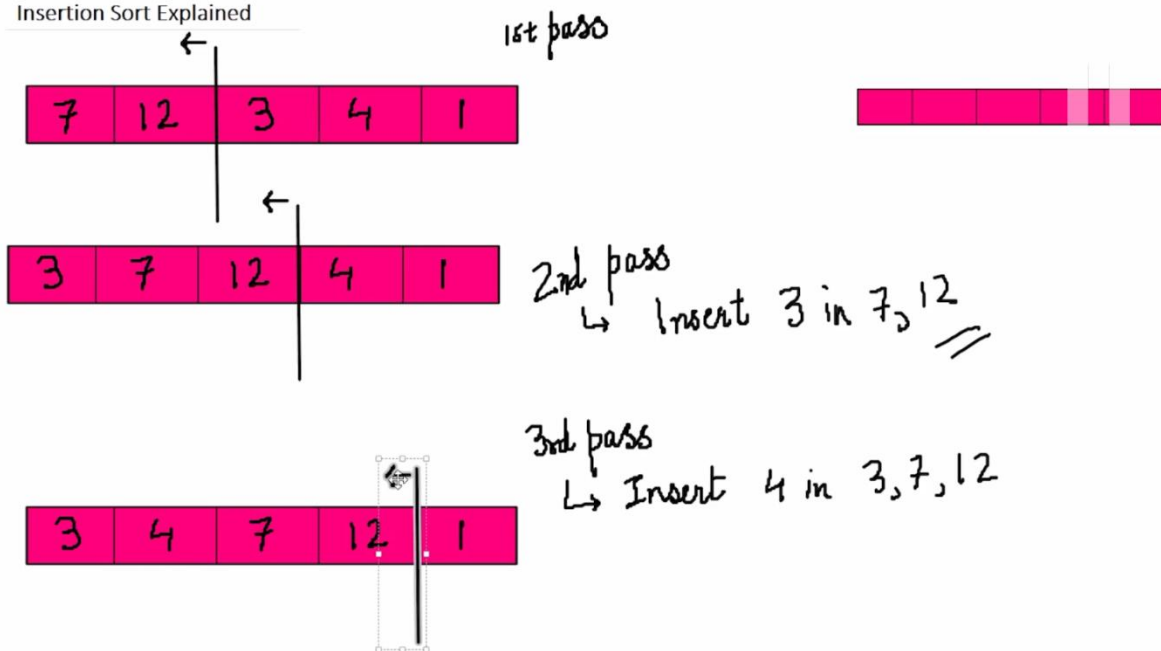
Insertion Sort Explained

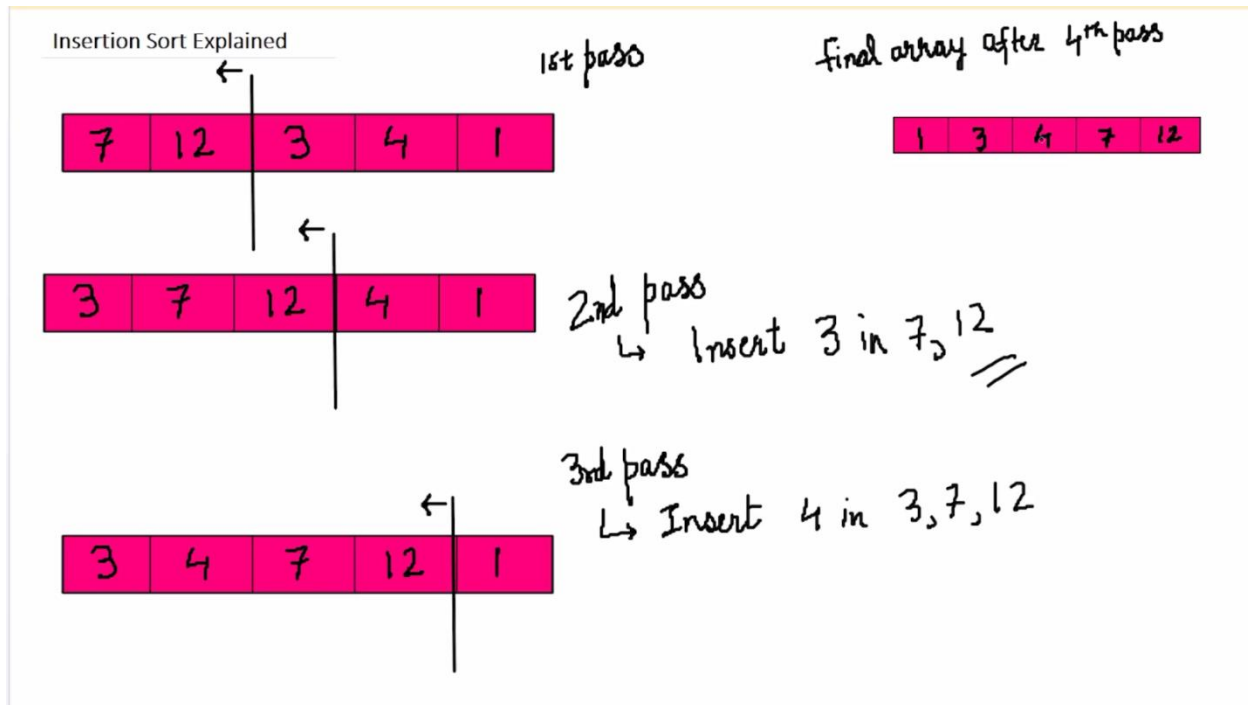


Insertion Sort Explained



Insertion Sort Explained





1. For number of elements in array = n
2. **Total no. of passes = $n - 1$**

No. of comparison in 1st pass = 1 = No. of possible swaps in 1st pass

No. of comparison in 2nd pass = 2 = No. of possible swaps in 2nd pass

No. of comparison in 3rd pass = 3 ... similarly for others

1. **No. of comparison in $(n-1)$ th pass = $n-1$ = No. of possible swaps in $(n-1)$ th pass**
2. **Total no. of swaps = $\frac{n(n-1)}{2} = 1 + 2 + 3 + \dots + n-1$**
3. **Time complexity of insertion sort = $O(n^2)$**
4. **Total no. of swaps in best case (when array is already sorted) = $n - 1 = O(n)$**
5. It is **Stable** algorithm
6. It is **Adaptive** (when the array is sorted) by nature
7. Its intermediate result is not useful