

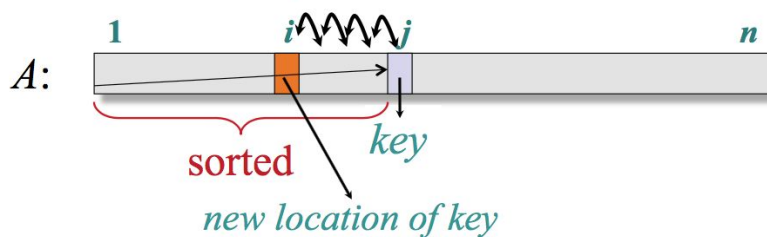
# Lec3\_Sorting

## Sorting

- Problem
  - Input : Array  $A[1 \dots n]$  of numbers
  - Output : permutation  $B[1 \dots n]$  of  $A$  such that  $B[1] \leq \dots \leq B[n]$
- Insertion Sort

**INSERTION-SORT** ( $A, n$ )  $\triangleright A[1 \dots n]$   
for  $j \leftarrow 2$  to  $n$   
    insert key  $A[j]$  into the (already sorted) sub-array  $A[1 \dots j-1]$ .  
    by pairwise key-swaps down to its right position

### Illustration of iteration $j$

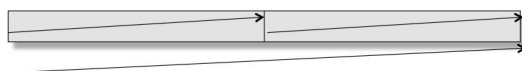


- Running Time
  - Normal :  $\Theta(n^2)$  compares
  - Binary Insertion :  $\Theta(n^2)$  swaps
- Merge Sort

divide and conquer

**MERGE-SORT**  $A[1 \dots n]$

1. If  $n = 1$ , done (nothing to sort).
2. Otherwise, recursively sort  $A[1 \dots n/2]$  and  $A[n/2+1 \dots n]$ .
3. “Merge” the two sorted sub-arrays.



**Key subroutine: MERGE**

- Running Time
    - Method
- Merge-Sort  $A[1 \dots n]$
1. If  $n = 1$ , done
  2. Recursively sort  $A[1 \dots \lceil n/2 \rceil]$  and  $A[\lceil n/2 \rceil + 1 \dots n]$

### 3. Merge two sorted list

- $\Theta(n \lg(n))$