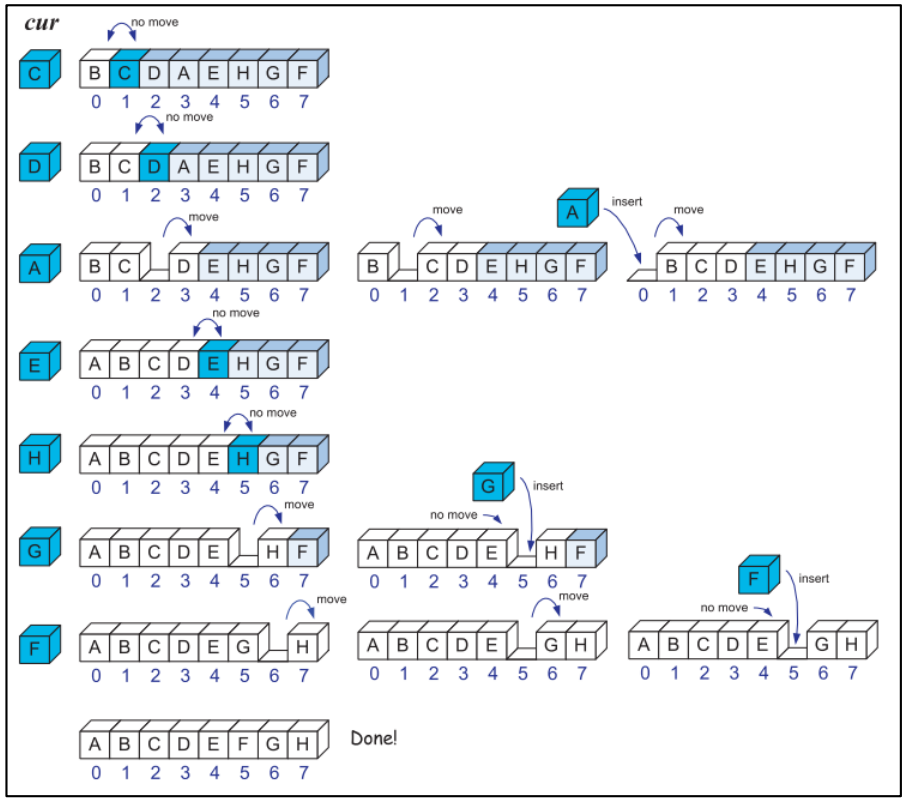
Insertion Sort

Jyh-Shing Roger Jang (張智星)
CSIE Dept, National Taiwan University

Insertion Sort

- Animation of insertion sort
  - Simple animation, Romania folk dance, electroacoustic
- Walk-through



Insertion Sort: Pseudo Code

- Pseudo code of insertion sort

Algorithm InsertionSort(A):

Input: An array A of n comparable elements

Output: The array A with elements rearranged in nondecreasing order

for $i \leftarrow 1$ to $n-1$ **do**

{Insert $A[i]$ at its proper location in $A[0], A[1], \ldots, A[i-1]$}

$cur \leftarrow A[i]$

$j \leftarrow i-1$

while $j \geq 0$ and $A[j] > cur$ **do**

$A[j+1] \leftarrow A[j]$

$j \leftarrow j-1$

$A[j+1] \leftarrow cur$ {cur is now in the right place}

Code Fragment 3.7: Algorithmic description of the insertion-sort algorithm.

Insertion Sort: C++ Function

- Real code in C++

```
void insertionSort(char* A, int n) {          // sort an array of n characters
  for (int i = 1; i < n; i++) {               // insertion loop
    char cur = A[i];                          // current character to insert
    int j = i - 1;                            // start at previous character
    while ((j >= 0) && (A[j] > cur)) {        // while A[j] is out of order
      A[j + 1] = A[j];                        // move A[j] right
      j--;                                    // decrement j
    }
    A[j + 1] = cur;                           // this is the proper place for cur
  }
}
```

Code Fragment 3.8: C++ code implementing the insertion-sort algorithm.

- Observations
  - Stable algorithm
  - To reduce comparisons ➔ Use binary search on the sorted part
  - To minimize movements ➔ Use another vector of pointers

Quiz for Insertion Sort

- Show each step of insertion sort on the vector:

  3 5 1 4 2 7 9 6 8