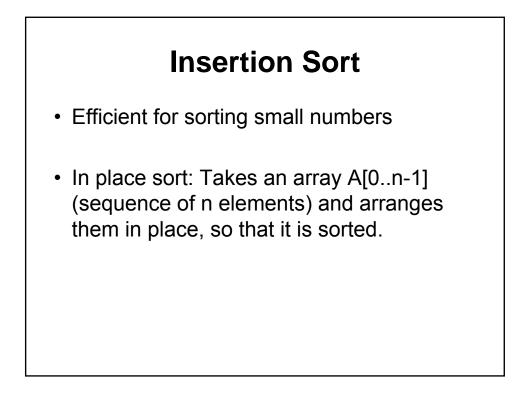
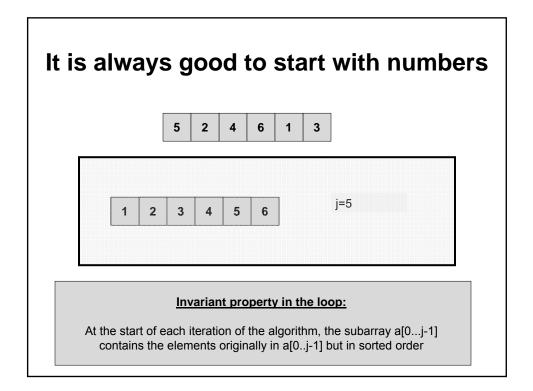
# Order Analysis of Algorithms

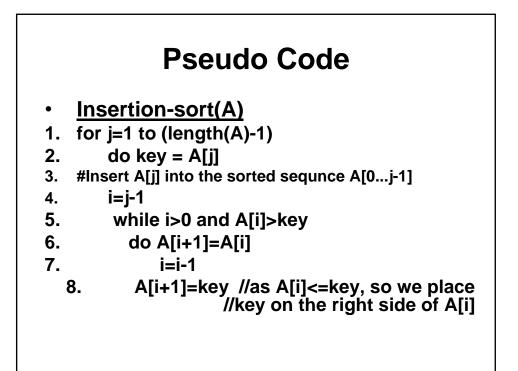
Debdeep Mukhopadhyay IIT Kharagpur



- Input: A sequence of n numbers, a1,a2,...,an
- Output: A permutation (reordering) (a1',a2',...,an') of the input sequence such that a1'≤a2' ≤... ≤an'
  - Comment: The number that we wish to sort are also known as keys







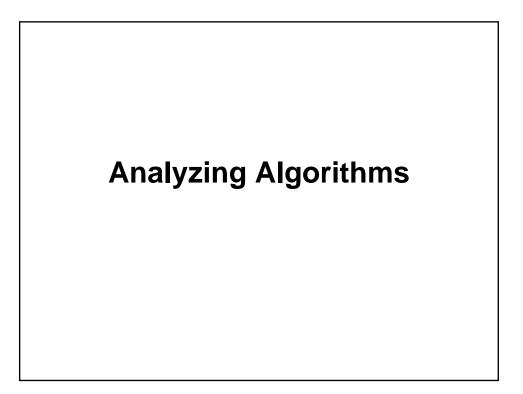
### Loop Invariants and Correctness of Insertion Sort

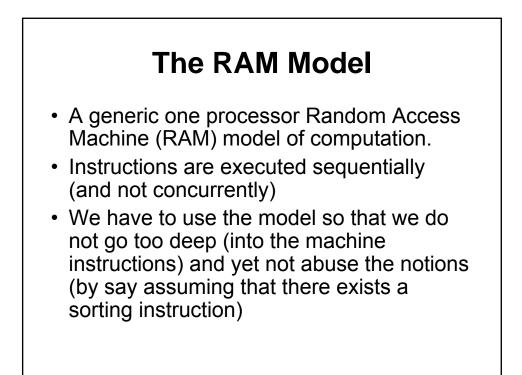
- **Initialization:** Before the first loop starts, j=1. So, A[0] is an array of single element and so is trivially sorted.
- **Maintenance:** The outer for loop has its index moving like j=1,2,...,n-1 (if A has n elements). At the beginning of the for loop assume that the array is sorted from A[0..j-1]. The inner while loop of the jth iteration places A[j] at its correct position. Thus at the end of the jth iteration, the array is sorted from A[0..j]. Thus, the invariance is maintained. Then j becomes j+1.
  - Also, using the same inductive reasoning the elements are also the same as in the original array in the locations A[0..j].

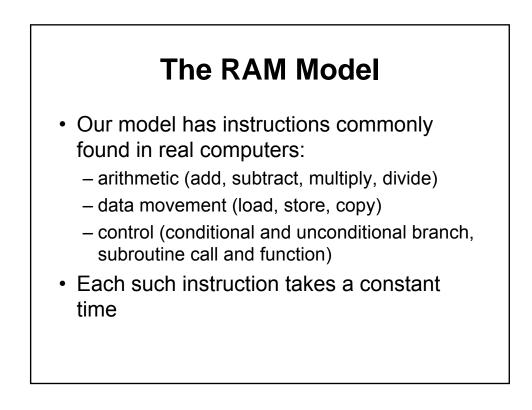
### Loop Invariants and Correctness of Insertion Sort

 Termination: The for loop terminates when j=n, thus by the previous observations the array is sorted from A[0..n-1] and the elements are also the same as in the original array.

Thus, the algorithm indeed sorts and is thus correct!







### Data types & Storage

- In the RAM model the data types are float and int.
- Assume the size of each block or word of data is so that an input of size n can be represented by word of clog(n) bits, c≥1
- $c \ge 1$ , so that each word can hold the value of n.
- c cannot grow arbitrarily, because we cannot have one word storing huge amount of data and also which could be operated in constant time.

### Gray areas in the RAM model

- Is exponentiation a constant time operation? NO
- Is computation of 2<sup>n</sup> a constant time operation?
  Well...
- Many computers have a "shift left" operation by k positions (in constant time)
- Shift left by 1 position multiplies by 2. So, if I shift left 2, k times...I obtain 2<sup>k</sup> in constant time !
  - (as long as k is no more than the word length).

### Some further points on the RAM Model

- · We do not model the memory hierarchy
  - No caches, pages etc
  - May be necessary for real computers and real applications. But the discussions are too specialized and we do use such modeling when required. As they are very complex and difficult to work with.
  - Fortunately, RAM models are excellent predictors! Still quite challenging. We require knowledge in logic, inductive reasoning, combinatorics, probability theory, algebra, and above all observation and intuition!



- The time taken to sort depends on the fact that we are sorting how many numbers
- Also, the time to sort may change depending upon whether the array is almost sorted (can you see if the array was sorted we had very little job).
- So, we need to define the meaning of the **input size** and **running time**.

## Input Size

- Depends on the notion of the problem we are studying.
- Consider sorting of n numbers. The input size is the cardinal number of the set of the integers we are sorting.
- Consider multiplying two integers. The input size is the total number of bits required to represent the numbers.
- Sometimes, instead of one numbers we represent the input by two numbers. E.g. graph algorithms, where the input size is represented by both the number of edges (E) and the number of vertices (V)

