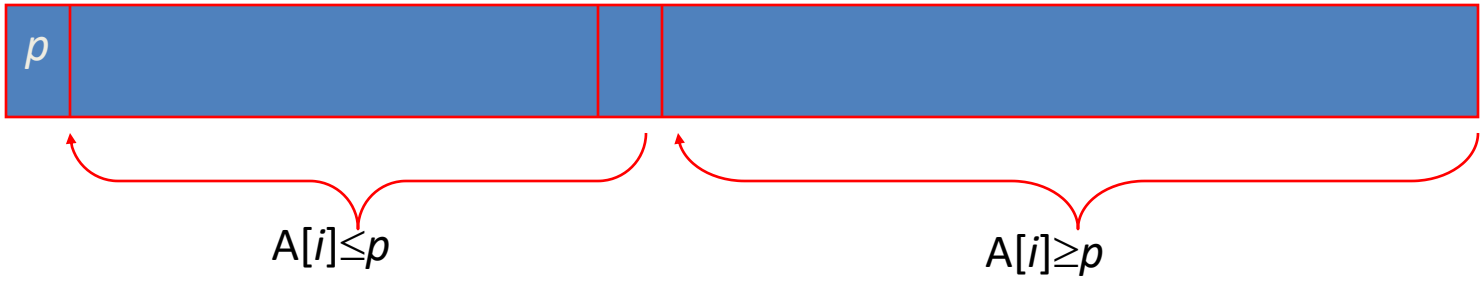
SNS College of Engineering Coimbatore - 641107

Quick Sort

Introduction

- Select a *pivot* (partitioning element) – here, the first element
- Rearrange the list so that all the elements in the first *s* positions are smaller than or equal to the pivot and all the elements in the remaining *n-s* positions are larger than or equal to the pivot (see next slide for an algorithm)



- Exchange the pivot with the last element in the first (i.e., ≤) subarray — the pivot is now in its final position
- Sort the two subarrays recursively

Algorithms

Algorithm *Partition*$(A[l..r])$
//Partitions a subarray by using its first element as a pivot
//Input: A subarray $A[l..r]$ of $A[0..n-1]$, defined by its left and right
// indices l and r $(l < r)$
//Output: A partition of $A[l..r]$, with the split position returned as
// this function's value
$p \leftarrow A[l]$
$i \leftarrow l$; $j \leftarrow r+1$
repeat
  repeat $i \leftarrow i+1$ **until** $A[i] \geq p$
  repeat $j \leftarrow j-1$ **until** $A[j] \cdot p$
  swap$(A[i], A[j])$
until $i \geq j$
swap$(A[i], A[j])$ //undo last swap when $i \geq j$
swap$(A[l], A[j])$
return j

Algorithm

5 3 1 9 8 2 4 7

2 3 1 4 5 8 9 7

1 2 3 4 5 7 8 9

1 2 3 4 5 7 8 9

1 2 3 4 5 7 8 9

1 2 3 4 5 7 8 9

Break

- **Assemble the Quotes**
- Preparation: For a group of 30 people, print 5 or 6 quotes or phrases on a paper (i.e. Face that launched a thousand ships; Fools rush in where angels fear to tread; Picture is worth a thousand words; Power corrupts; absolute power corrupts absolutely; etc.) and cut that printed paper so that each word of each phrase is a separate piece of paper. Fold up each of these 30 or so bits of paper and give one to each participant.
- Activity: When you say "go," have everyone simulatneously open their folded paper, then move around the room and find other words related to a possible phrase, from people in the room and try to complete the phrases. When they have feel they created a phrase, they can check in with the facilitator. This involves people to suddenly get energized, both in mind and body.

Analysis of quick Sort

- Best case: split in the middle — $\Theta(n \log n)$
- Worst case: sorted array! — $\Theta(n^2)$
- Average case: random arrays — $\Theta(n \log n)$

- Improvements:
  - better pivot selection: median of three partitioning
  - switch to insertion sort on small subfiles
  - elimination of recursion

  These combine to 20-25% improvement

- Considered the method of choice for internal sorting of large files ($n \geq 10000$)

Activity

- $\Theta(n \log n)$
- $\Theta(n^2)$
- $\Theta(n \log n)$