TA: Jade Cheng ICS 311 Quiz Solution #1 Jan 14, 2008

Quiz #1

List 7 classes of program complexity, in increasing order [4 pts]

Answer:			
	0(1)	Constant	(e.g, array access)
	O(lgn)	Logarithmic	(e.g, binary search)
	0(n)	Linear	(e.g, sequential search)
	O(nlgn)	nlogn	(e.g, mergesor)
	$O(n^2)$	Quadratic	(e.g, selection sort)
	$O(n^3)$	Cubic	(e.g, multiply 2 <i>nXn</i> matrices)
	$O(2^{n})$	Exponential	(e.g, travelling salesman)

What is the Θ complexity of insertion sort? Briefly justify your answer [3 pts]

Answer:

Insertion sort is a $\Theta(n^2)$ function.

For each item i, i = 2 to n, compare with sorted items one by one (maximum n - 1 items) and insert in the correct position. This involves $\Theta(n)$ comparisons, therefore the time complexity of insertion sort function is $\Theta(n^2)$.

What is the Θ complexity of merge sort? Briefly justify your answer [3 pts]

Answer:

Insertion sort is a $\Theta(nlgn)$ function.

The array holding *n* items is split into 2 equal (if possible, or shorter on the right side) pieces repeatedly, until only 1 or 2 items are in each piece. This process is $\Theta(lgn)$.

The pieces are merged together by comparing the first numbers in each segment, selecting the smallest (or largest depending on the sorting order), and putting that into the first position in the merged piece. This comparison and copy is done until all items are in the merged array. This procedure takes 1 comparison for a segment of 2 items, $2\sim3$ comparisons for 4 items, etc, up to n - 1 comparisons for *n* items (the last merging). This step takes $\Theta(n)$.

Therefore, merge sort is a $\Theta(nlgn)$ function.