CSE525 Lec1: Recursion

Debajyoti Bera (M21) <u>https://sites.google.com/a/iiitd.ac.in/cse525-m19</u>

Insertion Sort

| 26 54 26 54 | 4 93 | 17 | 77 | 31 | 44 | 55 | 20 | inserted 26 |
|----------------|------|----|----|----|----|----|----|-------------|
| | | 17 | 77 | 1 | | | L | 1 |
| | T | | | 31 | 44 | 55 | 20 | inserted 93 |
| 17 26 | 6 54 | 93 | 77 | 31 | 44 | 55 | 20 | inserted 17 |
| 17 26 | 6 54 | 77 | 93 | 31 | 44 | 55 | 20 | inserted 77 |
| 17 26 | 6 31 | 54 | 77 | 93 | 44 | 55 | 20 | inserted 31 |
| 17 26 | 6 31 | 44 | 54 | 77 | 93 | 55 | 20 | inserted 44 |
| 17 20 | 6 31 | 44 | 54 | 55 | 77 | 93 | 20 | inserted 55 |
| 17 20 | 0 26 | 31 | 44 | 54 | 55 | 77 | 93 | inserted 20 |

| Fix(i) : Assuming A[1i- A[1i]. $F_{ix}(5)$ where A? | [34,4,2,5] ? |
|--|---|
| Fix(5) shenAz JS(i) : Sorts A[1i] usi | $\frac{1}{3}, \frac{3}{4}, \frac{5}{2}$ |
| ALI InsertionSort(): $I_{S(n)}$ Is(i) IS(2) | JS(i-1) [JS(i) |
| IS(i): Rohum if i=0 or i>n | IS(N) |
| (IS(i): Pohum if i=0 or i>n > Receiver (i) Fix (i=1) LS(i=1) | 1,, i-of may be |
| (IS(i=1)) | $A= \left[3, 4, 1, 2, 5, 7 \right]$ |
| $T_{S}(i):$ or return if $i \leq 1$ | Trie |
| $J_{S}(i-i) // A [1-i-i]$ $J_{S}(i-i) // A [1-i-i-i]$ with be Fix(i) sorted | $\rightarrow \left(\begin{bmatrix} 1, 2, 3, 4, 5, 7 \end{bmatrix} \right)$ |
| Insertion Sort(): Is(n) |) |

Proof of Correctness

Fix(i) : Assuming A[1...i-1] is
sorted, sort A[1...i].

IS(i):

<u>ړ</u>. Return if 2· • ን د Fix <u>թություն</u> (<u>թություն</u>)

Claim: IS(i) sorts A[1...i] in place. Proof by induction on i. Base case: When i=0, the claim is trivially true since

- A[1...0] is an empty array.
- IS(0) returns immediately

Induction hypothesis: Assume that IS(i-1) sorts A[1...(i-1)] in place.

i=0 Induction statement: To prove that IS(i) sortsIS(i-1) A[1...i] in place, for i > 0.

IS(i) first calls IS(,,i-1). By IH, after this, A[1...(i-1)] is sorted.

IS(i) then calls Fix(i() which inserts A[i] inside the already sorted array A[1...(i-1)] in its right place in the sorted order. So after this call A[1...i] will be sorted.

Analysis of Insertion Sort

Assume that Fix(j) takes O(j) steps.

Let T(k) denote the time complexity of IS(k).

Write a recursive expression for T(k).

$$\Gamma(K) = \tau(K-1) + O(K) = O(K^{2})$$

Exercise

Fix(j) : Assume A[1...j] is sorted, except its last element. Place the min element of A[1...j] in the right place inside A[1...j].

Write a recursive algorithm for FixPosLast(A). Fig().

(Optional) Prove that your algorithm is correct using induction.

Can you speed up Fix(j) ? Either asymptotically or according to clock time.

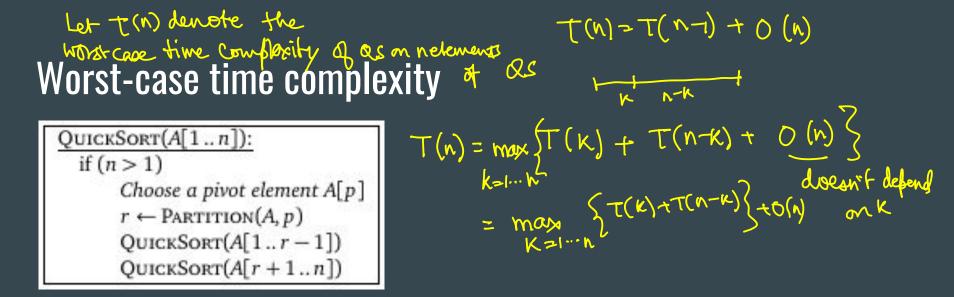
Is this correct?
IS(i):
 Return if i=0
 IS(i-1)
 FixPosLast(i)

Recursive Sort by Dividing at the Middle

- 1. Pre-process array.
- 2. Assume <u>left-half</u> is correctly sorted.
- 3. Assume **<u>right-half</u>** is correctly sorted.
- 4. Construct sorted version of the entire array.

Merge-sort
1. ? no pre-processing
2,3.
$$\checkmark$$
 Accurrence for
4. ? Merge ()
Time-complexity ? $T(n) = 2T(n/2)$
 $+O(n)$





What is the recurrence for T(n) in terms of r?

What is the recurrence for the worst-case expression for T(n)?