Time Complexity

Big O complexities Amount of time taken by algorithm to run as a function of length of input. (4 pol) 0: smit similaropos . E for lint i=0; i<N; i++)

(s'u)0: unit situations. A cout<<"Hello"; (E'M) O O'Cm) since } Time complexity order Note: It is not actually time but cell opens. (N 601)-0 SPACE COMPLEXITY : -(17)0 Amount of space taken by algorithm to run as a function of length of input. (190111) o (n) 0 Units to represent:-(EN)0 (i) Big O: Upper bound (But) 0(2") (ii) Theata O: Average case x2) for a toom - ("11) o (iii) omega 2: Lower bound end stand ripper bound Lower . A bound (isme & bko check wuga)

Biq 0: complexities: 1. constant time: 0(1) 2. Linear time: O(n) 3. Logarithmic time: O(log N) 4. Quadratic time: O(N^2) courie time: O(N13) "oller" & kuss Time complexity order: o (1) o that complexion for O(log N) 0 (III) -: KLIXJIANO) DE mount of space taken by algorithm (430 sun o (nlogn). testin for Hipped to noisein so 0(n2) O(n3) -: to refound :-0(zh) (rang) bound roff (Brey) O(NI) Maria B. Australia (Ontil O(Nn) - Most complex Lower & Cours Lowny

front broots