## Assignment - 1

## Section+A 1 314 John tony ! Dangarmon or of toly !

1. What are time Complexity and Space Complexity of an algorithm?

ans. Time Complenity measures the amount of time an algorithm takes relative to the input Size

· Space Complenity measurups the amount of memorry an algorithm uses relative to the input size.

2. How Substitution method works force solving recurrence?

In the Substitution method, we guess the solution and then prove it using mathematical induction. It involves substituting the guessed forum into the recurrence to verify if it satisfies the

3. What are the Conditions for which master method is not applicable?

· mæster method is not applicable when:

Aandard form:

T(n) = aT (n/b)+ flored one one

division is irraegular.

findion (mutos) ( part and some training the

Blacif (ann [mid]> May) high mid ath

it I birn tool

4. What is recurrence? What are the methods to Solve da recurrence? · A recurrence is an equation that defines a kunction based on its values out Smallore imputs sith of suited so reacher anthoropals on \* methods to solve! · Course Complement protection control of the size. Recursion Tree method top on promon privise master method of most of the dise wolf . Section-B 1. Write a recursive of non-receverative algorithm for bineray search and emplain why it is in more efficient that linears Search would · Receirestus Binary Search! int binaragearcon (int arcretz, int low, int high int key) for the total & of (low) high) rectures -1; int mid = (low+high) (2; if (order [mid] == key) return binary Search (arre, low, mid-1, key); else return binarysearch (arm, mid +1, high, key) Non-Recursive Binarry Search int binary Search (int arent], int n, int key-)? int low = 0 . high = n-1; pour ?? while (low <= high) { int mid = (low + high) /2: if [arar [mid] = = key) recturer mid; else if (arm [mid]) key) high = mid - 1; return -1; 1

· Efficiency: Binarry Search hoes O(logn) timo Complexity, while Linear Search hers O(m), making Binary Search Paster for large Sorded datasets.

2. Solve the following recurrence using Recursion Thee method. . . moth (dim) To = cont

· Each lovel does constant bookorck (1).

· Nembere of levels = log n.

· Total work = Ollogin).

(b) T(n) = 3T (n/4) + cn2 201/400 (200000)

· Tree root work - ch?

· Cach level reduces Size to 114, so work reduces faston.

. The dominant work comes from the react level: 1 O (n3). 2 m 20 to (in rab (n)),

. Colottons T(n): C(n).

## Section - c

1. Emplain Varcious Asympotic Notations that descreibe the running time of an algorithm.

. Blg() (0): Uppor bound (worest-case).

· Omega (sz): Lowers bound (best-cose).

· Theta (B): Tight bound (average-case).

· Little o (0): Structly less than a function.

· Little omega (w): Strictly greater than a function.

Extend : Divard Louren part ( (100) June 2. Complain moestern's method and solve the tollowing the currence using master method. - Masterce mothed Solves recurrence of the forem: Tm = at (n/b) +fcm. bother method (a)  $T(n) = 2T(n|u) + \sqrt{n} + (a|m) + Cont (a)$  (a = 2, b = 4, f(n) = n'/2· Comparce nogba = nogu = no.5 and fen. · Both arce no.5, thus case 2 of mostor's theorem applies. To - (1) 18 (1) (1) · Solution: T(n) = 0 (n° 5 Logn). (b) T(n) = 2T (n/2) + n3. (20) toon on a = 2, b = 2, f(n) = n3: notes = 2000 box · nugba a = plogo 2 min haming . f(n) dominates n. so case3 applies. · Solution: T(n) = O(n3). to fortain various. Asympotic thations that Section . (c) describe the receiving time of anolyprosent (cases). House period (marst. 1886). (Corredo (ca.): Lower bound (best case) · Thota (6): Tight butted (increase (656)). · Little o(0): shidtly less than a landion. , little cinege (10): Shidhy grander than a . (Witoru)