Q 1

Ans

1) Asymptotic Defines that a function is tending to one axis you can say like it is a limit Example of Asymptotic function could be e^2.

2) Asymptotic notation in Data Structures are used to Represent time Complexities of your Algorithm.

3) It tells about the worst, average and Best Case time Complexity i.e time taken to execute a particular algorithm or Code.

4) The denotions or Best, worst and average case are :

O (Big 0) → worst case

Ω (Omega) → Best Case

Θ (theta) → Average Case.

3

5) But In data structure we Always Consider that what is the worst case of An Algorithm or Code that define that how Much Computational power Computer Consumes to Execute.
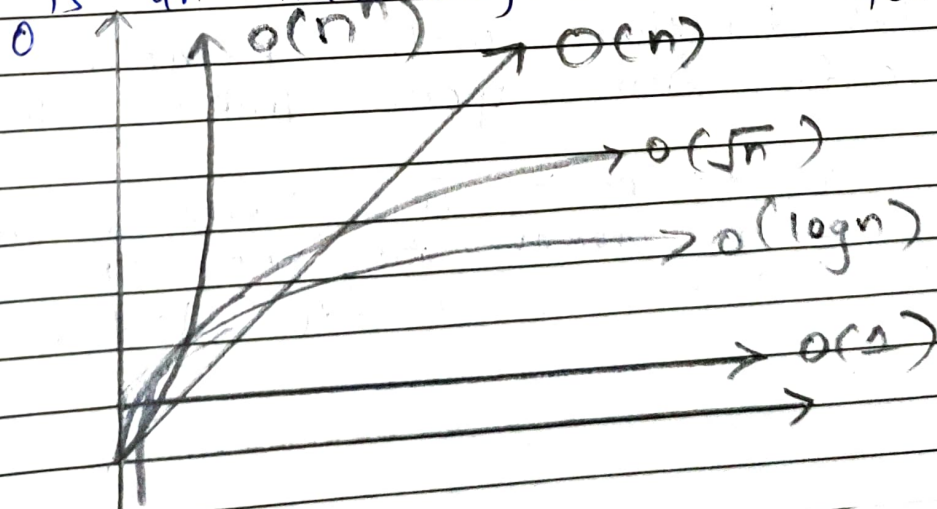
6) Ex: int Count = 0
   for (int i = 0 ; i < n ; i++) {
       Count ++;
   }

   In this Code the loop has time complexity of $O(n)$ and the Count variable updation has $O(1)$
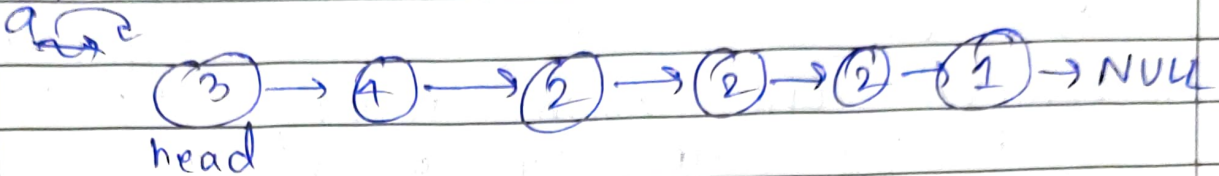   Time Complexity = $O(n)$

7) $O(1) \Rightarrow < O(\log n) < O(\sqrt{n}) < O(n)$
   $< O(n^2) < O(n^3) \ldots < O(2^n) < O(n!)$
   $< O(n^n)$
   This is an increasing order (growth func)

Question Nos.		Marks Awarded

4)   Current linked list :→ a

a→a→a
    ③ → ④ → ② → ② → ② → ① → NULL
head

5)   Two pointer Approach to Remove Duplicates :

RemoveDuplicate

~~ListNode~~ ~~ListNode~~ (struct node* a) {

    struct node* t1 = head;
    struct Node* t2 = head→next;
    while (t1 != NULL && t2 != NULL) {
      IF (value of t1 == value of t2) {
       while (t2 != t1) {
        t2 = t2→next;   free(t2);
        } t1 = t2;
        t2 = t2→next;
      }
      t1 = t1→next;
      t2 = t2→next;
    }

logic : Using Two pointer t1 and t2
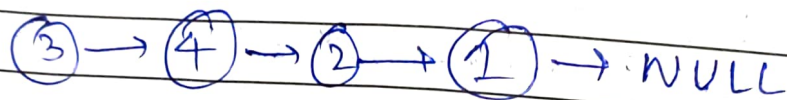Checking if their values one
Same If not then ~~t~~ move t1
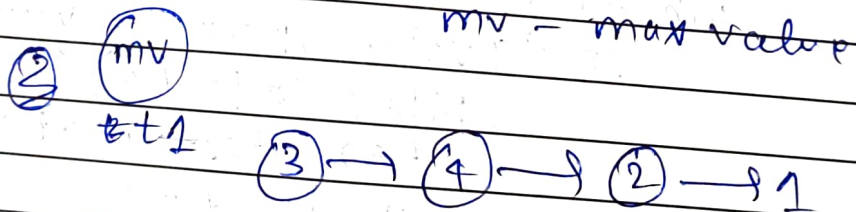and t2 to next node If
t1.val == t2.val then traverse t2
Using while loop till the ~~e~~ t1 != t2
and free (t2) Node Every time

then liking the current t_2
to t_2 so that the Duplicate
Remove and again moving t_2 to
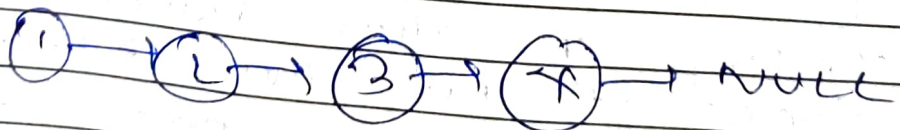next the startting again checking.

6) The linked list would be

$3 \rightarrow 4 \rightarrow 2 \rightarrow 1 \rightarrow NULL$

7) Sorting this final list using
two Node by given them
max value.

mv - max value

② (mv)
t_1
$3 \rightarrow 4 \rightarrow 2 \rightarrow 1$

(mv)
t_2

Checking the value and
Componing and changing the
list

$1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow NULL$

Name : Jeet Doshi
SAP : 60009230007
Sub : DS
Roll NO : D038

Q.3

Ans

$$(A + B * (C - D)^E) / (F - (G + H * I))$$

starting from left

Input	op stack	Postfix
(	(	
A	(	A
+	(+	A
B	(+	A B
*	(+ *	A B
(	(+ *	A B
C	(+ *	A B C
—	—	A B C * + (
D	—	A B C * + (D
)	—)	A B C * + (C D
^	—) ^	A B C * + (C D
E	—) ^	A B C * + (C D E
)	—) ^)	A B C * + (C D E
/ C	C + * C	