BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI
(END SEMESTER EXAMINATION)

CLASS: BTECH **SEMESTER : IV**
BRANCH: BT/CHEMICAL/ECE/EEE/MECH/PIE **SESSION : SP/2024**

SUBJECT: CS261 FUNDAMENTALS OF DATA STRUCTURES AND ALGORITHM

TIME: 3 Hours **FULL MARKS: 50**
INSTRUCTIONS:
1. The question paper contains 5 questions each of 10 marks and total 50 marks.
2. Attempt all questions.
3. The missing data, if any, may be assumed suitably.
***4*. Before attempting the question paper, be sure that you have got the correct question paper.**
5. Tables/Data hand book/Graph paper etc. to be supplied to the candidates in the examination hall.

CO BL

Q.1(a) Consider the following algorithm [5] [1] 3

```
Algorithm Test(int n)
{
value=0;
 for (i =n; i>0;i/=2)
          for j in range(0 to i)
                    value=value+1
return value
 }
```

Write down the stepwise demonstration for the time complexity of the above. Also discuss whether it is valid or not to say $\mathbf{f(n) = O(n^2)}$ if f(n) is the time complexity of the above algorithm. Justify your answer.

Q.1(b) Consider we have two lists of arrays; one is sorted, and another is unsorted. There are anagram words that exist in the list [Anagram words are formed by rearranging the letters of another words, using the letters exactly once]. Anagram list are e.g. ‘listen’ and ‘silent’. Consider two scenarios where in one scenario the given list is sorted and, in another scenario, list is not sorted. Write an algorithm to find out the anagram in both scenarios. Discuss the computational complexity of the solution. [5] [2] 2

Q.2(a) The following sequence of operations is performed on stack. PUSH (40), POP, PUSH (5), PUSH (4), PUSH (5), POP, POP, PUSH (9), POP, POP. Demonstrate each step with a suitable diagram. Consider the following infix expression: [5] [2] 4
A+(B*C-(D/E^F)*G)*H.
Fill the below table depending on the conversion from given infix to postfix.

Symbol Scanned	Stack	Expression P
A	(	A
——	——	——
——	——	——
——	——	——
D	(+(-(	ABC*D
….	….	…
….	….	…
…	….	….
….	….	ABC*DEF^/G*-H*+

PTO

Similarly, another table is filled for the prefix notation for the same infix expression as follows. Fill the missing part of the table.

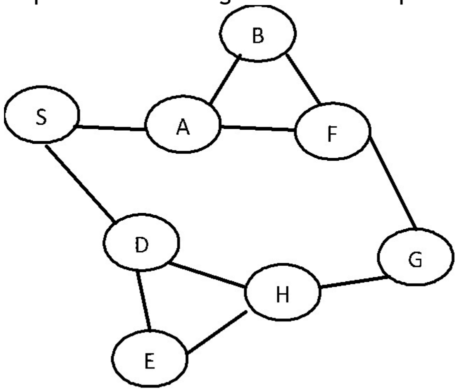
Symbol Scanned	Stack	Expression P
A	(	A
——	——	——
——	——	——
——	——	——
D	(+(-(	ABC*D
....	....	...
....	....	...
...	....	....
....	....	+A*-*BC*/D^EFGH

Q.2(b) Describe the steps of implementing Queue using minimum number of stacks, making PUSH operation costly. Consider another scenario where pop operation is made costly and write down the steps for the same to implement queue using stack data structure. [5] [4] 1, 2

Q.3(a) Create a linked list that consists of integers. Write an algorithm to insert a new element between any pair of consecutive elements. The new element is the average of any two consecutive integers. For example, if the list is
12 --> 34 ---> 56 --> 78 --> 15,
then the average of two consecutive integers 12 and 34 is 23 should be inserted between 12 and 34. The average of 34 and 56 is 45, which should be inserted as shown below:
12 --> 23 --> 34 --> 45 --> 56 --> 36 --> 16.
Write down the complexity of the algorithm. [5] [3] 6

Q.3(b) Compare the linked list and array to discuss the preference among the two in terms of insertion, deletion, searching, and sorting. Write down the advantage and use of doubly linked list and circular linked list. [5] 1, 2

Q.4(a) Consider the following graph where node S is the source and node G is the destination. Apply depth-first search and breadth-first search algorithm to reach the goal and demonstrate each step. Design an algorithm count() to find the number of nodes in Depth First Search generated at depth 3. [5] [3] 4



Q.4(b)	Consider the given pre-order and in-order traversal of the elements. Draw a Binary Tree using the given elements. Pre-order: K A M C B Y P F H In-order: M A B C K Y F P H Write down the post-order traversal of the tree formed using above given in-order and preorder traversal. Also, draw the binary search tree for the following elements and demonstrate each step to insert the element one by one in the binary search tree. 20,8,22,4,12,10,14,1,30	[5]	[3]	5
Q.5(a)	Consider a list of integer where the list is large. Which sorting algorithm is best in this scenario. Few more elements have been added at random index locations that messes up the sorted sequence. Which search procedure is considered as efficient to find an element in this scenario and why? If additional memory requirement is not considered, then which sorting algorithm is suitable after addition of random elements in the sorted sequence? Justify your answer.	[5]	[5]	6
Q.5(b)	Consider a list L having 'n' elements and partially sorted as $L[1,2,\ldots d_i-1,d_i,d_{i+1}\ldots,n]$, where $d_{i-1}>=d_i$ and $d_i<=d_{i+1}$. L={2,3,4,28,12,34,56,66,42,}} Based on the data given above suggest a suitable sorting algorithm to arrange the elements in ascending order. What would be the total number of iteration required to get the sorted list. Demonstrate the stepwise procedure.	[5]	[4]	3

::::::30/04/2024::::::M