Seat No.:

MIT ACADEMY OF ENGINEERING

Course Code : CS201 **July – 2017**

S. Y. B. Tech Examination
Semester – III
Cycle – II
In – Course Examination

DATA AND FILE STRUCTURES

Time: 2 Hours **Max. Marks : 50**
Total No. of Questions: 05 **Total No. of Printed Pages: 02**

Instruction to Candidates:

1. Assume suitable data wherever necessary
2. Non programmable scientific calculators are allowed
3. Black figures to the right indicate full marks

1. (a) Consider the complexity function *F(n)*, which measures the number of times POS and MAX are updated in step 3, in following algorithm, **[7] CO-2 L2**

Algorithm: Given a nonempty array INFO with N numerical values.

1. [Initialize.] Set K:=1, POS:=1, and MAX:=INFO[1].
2. Repeat steps 3 and 4 while K<=N:
3. If MAX< INFO[K], then:
   Set POS:=K and MAX:=INFO[K]
   [End of If Structure.]
4. Set K:=K+1.
   [End of Step 2 loop.]
5. Write: POS, MAX.
6. Exit.

Describe with examples and find

I. *F(n)* for worst case.
II. *F(n)* for best case.
III. *F(n)* for the average case when *n* =3 elements, assuming all arrangements of the elements in array INFO are equally likely.

[F(n) for Worst case with Example – 2 Marks, F(n) for Best Case with Example - 2 Marks, F(n) for the average case -1 Marks, and all arrangement of 3 elements – 2 Marks]

(b) Calculate frequency and total steps for, **[3] CO-2 L2**

Algorithm: SUM(X,Y,Z, m,n)

```
{
 for i:=1 to m do
      for j:=1 to n do
          Z[i,j]:=X[i,j]+Y[i,j];
}
```

[Total steps table – 2 Marks, Frequency – 1 Marks]

2. Suppose an insurance company keeps a linear array YEAR(from 1950:2000) such that YEAR[K] contains the number of customers born in year K. Write algorithms for each of the following tasks: **[10] CO-1 L3**

(a). To print each of the years in which no customer was born.

(b). To find the count of number of years in which no customer was born.

(c). To find the number of customers who will be at-least 50 years old at the end of the current year 2017.

[INPUT, OUTPUT, initial conditions and title of three algorithms – 3 Marks, Algorithm (a) - 2 Marks, Algorithm (b) – 2 Marks, Algorithm (c) – 3 Marks]

3. In a linked list of students, each node contains NAME and DFS_MARKS. Write algorithm to insert and delete STUDENT in the sorted linked list. Linked list is sorted according to DFS_MARKS. **[10] CO-1 L3**

[INPUT and OUTPUT- 1 Marks, Data Type/ Class Node- 1 Marks, Data Type/ Class Linked list with its operation – 2 Marks, Algorithm to Insert in sorted linked list -3 Marks, Algorithm Delete from sorted linked list -3 Marks]

4. Write a program to test if a string is palindrome using singly linked representation of the stack. **[10] CO-3 L3**

[Data Type Stack using singly linked representation -1 Marks, Push Function with status checking - 3 Marks, POP Function with status checking - 3 Marks, Main Logic for a palindrome using stack - 3 Marks]

5. (a) Evaluate the following postfix notation of expression(Show status of stack after execution of each operation): 4, 10, 5, +, *, 15, 3, /, - **[5]** **CO-3** **L2**

*[Evaluation for every operator with stack status -, +, *, / - 4 Marks, Final status of stack and result – 1 Marks]*

(b) Consider the following queue of character, where queue is circular array which is allocated 6 memory cell. FRONT=2, REAR=4, **[5]** **CO-3** **L2**

QUEUE = _, *Ahmedabad, Kochi, Delhi*, _ , _.

Describe the queue using FRONT and REAR update operations for following sequence of operations,

(1) *Indore* is added to the queue.

(2)Two cities are deleted.

(3)*Jaipur, Pune, M*umbai are added to the queue.

[Queue FRONT and REAR update operations – 2 Marks,
Indore is added to the queue with status – 1 Marks, Two cities are deleted with status – 1 Marks, (3)Jaipur, Pune, Mumbai are added to the queue with status – 1 Marks.]