NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA

■ CS2005 Data Structures and Algorithms ■ Mid Sem Autumn 2019 ■ UG ■ Pg 1 ■ FM 30 ■ 2 Hrs ■

■ Answer ALL questions. ■ All parts of a question MUST be answered together. ■ Mere answers without proper explanation will not fetch marks. ■ Variation in quality of answer will vary the secured marks.

1. (a) Let S be a stack of size $n + 1$. Starting with the stack bearing first n natural numbers (highest number on top of stack), three elements are popped if top of stack is even, and one element (the number next to the one in top of stack) is pushed if top of stack is odd. These operations continue until the stack is empty or there are insufficient number of elements so that three pops cannot be done.
(i) How many elements will be finally left in stack when three elements cannot be further popped out due to insufficient elements in the stack?
(ii) Represent the number of `push` and `pop` operations required until end of the given process as functions of n. [$1 + 3$]
(b) A single array $A[1...\texttt{MAXSIZE}]$ is used to implement two stacks. The two stacks grow from opposite ends of the array. Variables `top1` and `top2` point to the location of the topmost element in stack starting from location 1 and stack starting from location `MAXSIZE` respectively. If the space is to be used efficiently, what is the condition for "stack full" for any of these two stacks? [2]
(c) Implement a stack (and its functions: `push` and `pop`) using one or more queues and functions defined on queue (`enqueue` and `delqueue`). Consider the situation where no other data structure like arrays, linked list is available for usage. [4]

2. (a) Represent the following sparse matrix `A` given below using triplet and orthogonal chain. Assume that index value starts from (1, 1) and pointer field occupies 4 bytes. Compute the total space required for each representation assuming 2 bytes of memory to store an integer. [$1 + 1 + 1 + 1$]

$$\begin{bmatrix} 0 & 0 & 3 & 0 & 4 \\ 0 & 0 & 5 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 6 & 0 & 0 \end{bmatrix}$$

(b) Convert the following infix expression to postfix expression using stack (consider the operands to have conventionally defined precedence and associativity).
$4 + 8 \times 5 / (7 - 6) + 2 \uparrow 3$
Evaluate the resultant postfix expression using stack. Show the stack content and intermediately formed postfix expression of each step clearly. [3.5]
(c) Consider a circular queue `Q` implemented using an array of 5 elements. Initially `Q` is empty. Show the content of `Q` and position of Front and Rear variables initially and after each of the following operations: (i) `Delete()`, (ii) `Insert(A)`, (iii) `Insert(B)`, (iv) `Delete()`, (v) `Insert(C)`, (vi) `Insert(D)`, (vii) `Insert(E)`, (viii) `Insert(F)`, (ix) `Insert(G)`, (x) `Delete()`. [2.5]

3. (a) What are the time complexities of finding 8th element from beginning and 8th element from end in a singly linked list? Let n be the number of nodes in linked list, you may assume that $n > 8$. [2]
(b) What is time complexity of `fun()`? [2.5]

```
int fun(int n)
{
int count = 0;
for (int i = n; i > 0; i /= 2)
for (int j = 0; j < i; j++)
count += 1;
return count;
}
```

(c) Let `P` be a singly linked list. Let `Q` be the pointer to an intermediate node `x` in the list. What is the worst-case time complexity of the best known algorithm to remove the content of the node `x` from the list? Elaborate the process. [2.5]
(d) An algorithm consumes $2^n + n^2$ units of running time for input size n. Derive the complexity of the algorithm with respect to O, Θ, Ω notations. [3]