NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA
■ Data Structures and Algorithms ■ CS-2005
■ Autumn–2022, Mid Sem ■ 30 Marks ■ 2 Pages
■ Time: 2hrs ■ Answer ALL Questions

---

1. (a) Find the worst case time complexity for the following operations on a singly linked list of length `n` with a HEAD and a TAIL pointer pointing to the first and last node respectively. Briefly justify your answers. [4]
   - Delete the k^{th} node from the beginning.
   - Delete the k^{th} node from the end.
   - Delete the middle element.
   - Delete the last element.

   (b) Search for item 57 in the following sorted array of elements using binary search. Show each step clearly. [2]
   07, 17, 27, 37, 47, 57, 67, 77, 87, 97

   (c) If $f_1(n) = \Omega(g_1(n))$ and $f_2(n) = \Omega(g_2(n))$, then $f_1(n) \times f_2(n) = \Omega(g_1(n) \times g_2(n))$. Justify if this statement is true or false. [2]

   (d) Enumerate advantages of circular list over grounded list. Calculate the number of pointers in a doubly linked list containing 10 nodes. [2]

2. (a) Considering the given precedence table with 3 being the lowest and 1 being the highest precedence, convert the following infix expression to Reverse-Polish notation using stack. The Each step must be shown clearly. [5]
   "$a + ((b - c)\hat{}d - e/f + (g * h + i/j))$"

Operator	Precedence
$\wedge$	3
$/,\ *$	2
$+,\ -$	1

   (b) Natural numbers are to be inserted to and deleted from a circular queue, which is implemented using an array of size 10 with the following rules. [5]
   i. Numbers are inserted in increasing order starting from 1.
   ii. Four insertions (numbers 1, 2, 3, and 4) are performed followed by two deletions.
   iii. The previous step is repeated three more times starting from number 5.

   Show the content of the circular queue, positions of FRONT and REAR after each operation.

   Write the generic overflow and underflow condition for a circular queue implemented using array.

3. (a) You are given an array `s[]` of length `n` and a procedure `reverse(s, i, j)` which reverses the order of elements in `s` between indices `i` and `j` (both inclusive). The following three lines of code are written to rotate `s` right by `n-k` positions. Find the missing arguments in the function calls. Explain the solution with an example array `s=[10, 20, 30, 40, 50, 60, 70, 80]`. Indexing starts from 1. [3]

```
reverse(s, ?, ?);
reverse(s, ?, ?);
reverse(s, ?, ?);
```

(b) Each value of a *priority queue* data structure is associated with a key. An entry in such a queue is a pair $< \texttt{value}, \texttt{key} >$. Priority queue can be implemented using two-dimensional array, where the number of rows equal to the number of priorities. Some of the common operations on *priority queue* are —

i. `insert(v, k)` inserts value `v` with priority number `k` in appropriate position

ii. `delete()` returns and deletes the entry with smallest key from priority queue

iii. `size()` returns the number of entries in the priority queue

iv. `isEmpty()` returns TRUE if the priority queue is empty otherwise FALSE

Determine the asymptotic complexity of the above four operations. Corroborate your answers with adequate explanation. [4]

(c) How much memory is required to store the sparse matrix given below using the following methods if an integer consumes 2 bytes and an address consume 4 bytes. [3]

$$S = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 6 & 8 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \end{pmatrix}$$

i. 2D array to store all the elements

ii. Triplet array

iii. Single chain linked list

iv. One chain per row

v. One chain per column

vi. Orthogonal list