DEPARTMENT OF INFORMATION TECHNOLOGY, NITK SURATHKAL

MID-SEMESTER EXAMINATION, NOVEMBER 2022

IT202: DATA STRUCTURES AND ALGORITHMS - I

Class: III SEM B.TECH. (IT) **Time: 1.5 Hr.**
Date: 02/12/2022 **Marks: 30**

Register No. | 2 | 1 | 1 | 0 | 0 | 9 | 3 | |

NOTE: 1. Answer all questions

2. All the subcomponents of the same question should be answered together

---

1. Evaluate the following *postfix* expression using stack showing all the intermediate steps.
6 2 3 + - 3 8 2 / + * 2 ^ 3 +

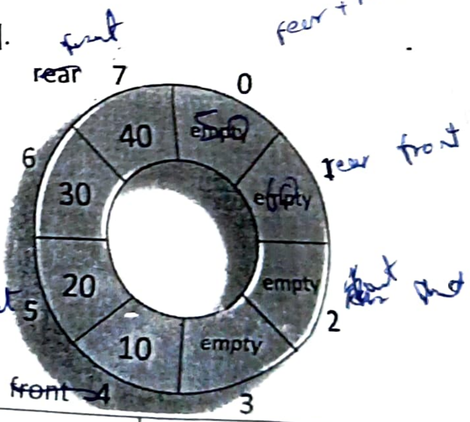
+8 ✓ (5)

2. Convert the following *infix* expression to an *prefix* expression using stack showing all the intermediate steps. Parenthesis optimization is mandatory. infix + postfix
((A + B) * C – (D – E)) * (F + G)

(5)

3. Consider an array-based *circular queue* of size 8 [index: 0 - 7]. The initial *queue* has four elements as shown: 10, 20, 30, and 40. The *front* and *rear* index are 4 and 7 respectively. Show the changes in the *circular queue* (changes of *front* and *rear* index positions and the respective values) after each of the following queue operaricns in the following table.

enQueue(50), *deQueue*( ), *deQueue*( ), *deQueue*( ), *enQueue*(60), *deQueue*( ), *deQueue*( ), *deQueue*( ), *enQueue*(70), *deQueue*( )



Operation	*Rear* index	Key value at *rear* index	*Front* index	Key value at *front* index
-	7	40	4	10
enQueue(50)				
deQueue( )				
deQueue( )				
...				

+ * (10)

((A+B)*C − (D−E)) *(F+G)

(13-4)

18

9 x 9

4. Consider an array-based HASHTABLE of size 10 [index: 0 - 9]. Insert the following keys into the hashtable using ***linear probing*** and ***double hashing*** scheme.

96, 48, 63, 29, 87, 77, 48, 65, 69, 93

For the ***linear probing*** and ***double hashing*** scheme, consider the following hash function:

$h_1(k) = k \bmod 10$

For the ***double hashing*** scheme, consider the following hash function for ***offset*** calculation:

$h_2(k) = 13 - (k \bmod 13)$

Show the changes in the below table for processing each of the keys:

For ***Linear probing***

no. of spaces you skipping

no. of position

Key	**h_1(k)**	**Offset value**	**Number of probes**
96			
48			
63			
29			
87			
77			
48			
65			
69			
93			

For ***Double hasing***

Key	**h_1(k)**	**Offset value**	**Number of probes**
96			
48			
...			

If you encounter an infinite loop at any point, you may quit and describe the probable reason.

(5 × 2)

65
78
91

5

6 +(13 - 9)

7 +(13 - 12)

6!

8

probe (probe+1) / mode

19 - (

(14) (6+4) | 10

(h₁(k) + 1) / size

13
26
39

0

13
26
39

87

9

65
78

9

(7 +1)
%10

67
78
91

8

52

12

4

9

13-9

4