Take Home Assignment 2019

18000768

May 13, 2019

Problem

A giant library has just been inaugurated this week. It can be modeled as a sequence of N consecutive shelves with each shelf having some number of books. No, think of the following two queries which can be performed on these shelves.

- Change the number of books in one of the shelves
- Obtain the number of books on the shelf having the kth rank within the range of shelves

A shelf is said to have the kth rank if its position is k when the shelves are sorted based on the number of the books they contain, in ascending order. Can you write a program to simulate the above queries?

Input Format

The first line contains a single integer T, denoting the number of test cases.
The first line of each test case contains an integer N denoting the number of shelves in the library.
The next line contains N space separated integers where the ith integer represents the number of books on the ith shelf where 1<=i<=N.
The next line contains an integer Q denoting the number of queries to be performed. Q lines follow with each line representing a query
Queries can be of two types:

- 1 x k - Update the number of books in the xth shelf to k (1 <= x <= N).
- 0 x y k - Find the number of books on the shelf between the shelves x and y (both inclusive) with the kth rank (1 <= x <= y <= N, 1 <= k <= y-x+1).

Output Format

For every test case, output the results of the queries in a new line.

Algorithm

```
Step 1:  Get testcases as T
Step 2:  Set I = 0
Step 3:  Repeat 4 to 15 steps while I <T
Step 4:      Get no of Shelves as S
Step 5:      Set J = 0
Step 6:      Repeat 7 to 8 steps while J <S
Step 7:          Get no of Books as B
Step 8:          Insert B to linked list
             [End of Loop]
Step 9:      Get no of queries as Q
Step 10:     Set J = 0
Step 11:     Repeat 12 to 14 steps while J <Q
Step 12:         Get query type as Qtype
Step 13:         if{Qtype = 1}
                     Get x,k
                     Update xth node with k in linked list
                     [End if]
Step 14:         if{Qtype = 0}
                     Get x,y,k
                     array = Insert data in xth to yth nodes in the linked list
                     Sort array
                     Print (k-1)th position in the array
                 [End if]
             [End of Loop]
Step 15:     Delete all Data Data in linked list
         [End of Loop]
```

Method to Insert data to linked list

```
Step 1: IF {AVAIL = NULL}
            Print OVERFLOW
            Go to Step 13
        [End IF]

Step 2: set AVAIL = AVAIL ->next

Step 3: newnode ->next = NULL

Step 4: newnode ->data = VAL

Step 5: IF(start == NULL)

Step 6:     start = newnode

        ELSE

Step 7:     ptr = start

Step 8:     Repeat step 9 while (ptr ->next != NULL)

Step 9:             ptr = ptr ->next

            [End of Loop]

Step 10:    ptr ->next = newnode

        [End IF]

Step 11: EXIT
```

Method to Update xth node with k in linked list

```
Step 1: Get x,k
Step 2: IF(start != NULL)
Step 3:     ptr = start
Step 4:     set I = 1
Step 5:     Repeat 6,7 steps while (I <x AND ptr ->next != NULL)
Step 6:         ptr = ptr ->next
Step 7:         I = I + 1
            [End of Loop]
Step 8:     ptr ->data = k
        [End IF]
Step 9: EXIT
```

Insert data in the linked list to an array and Sort them

```
Step 1: Get x,y,k
Step 2: IF(start != NULL)
Step 3:     ptr = start
Step 4:     set I = 1
Step 5:     Repeat 6,7 steps while (I <x AND ptr ->next != NULL)
Step 6:         ptr = ptr ->next
Step 7:         I = I + 1
            [End of Loop]
Step 8:     set I = 0
Step 9:     int array[100]
Step 10:    Repeat 11 to 13 steps while ( I <(y-x+1) )
Step 11:            array[I] = ptr ->data
Step 12:            ptr = ptr ->next
Step 13:            I = I + 1
              [End of Loop]
Step 14:      set I = 0,J = 0,Size = y-x+1
Step 15:      Repeat 16 to 19 steps while ( I <Size )
Step 16:            Repeat 17 to 18 steps while ( J <(Size - I) )
Step 17:                  IF (arr[I] >arr[J+i])
                                temp = arr[I]
                                arr[I] = arr[J+I]
                                arr[J+I] = temp
                  [End IF]
Step 18:                  J = J +1
                    [End of Loop]
Step 19:            I = I +1
              [End of Loop]
           [End IF]
Step 20:  EXIT
```

Delete all data in the linked list

```
Step 1: IF(start != NULL)
Step 2:     IF(start ->next != NULL)
Step 3:         ptr = start ->next
            [End IF]
Step 4:     start = NULL
Step 5:     Repeat 6 to 8 steps while ( ptr ->next != NULL )
Step 6:         preptr = ptr
Step 7:         ptr = ptr ->next
Step 8:         FREE(ptr)
            [End of Loop]
        [End IF]
```