

Arrays

Question 1:

There is a programming language with only four operations and one variable (X).

The four operations which can be performed are:

- i) X++
- ii) ++X
- iii) X—
- iv) —X

X++ and ++X both increments the value of X by 1.

X— and —X both decrements the value of X by 1.

The initial value of X is 0.

Given an array of strings “operations” containing a list of operations, write a program to print the final value of X when:

- a) operations = {“++X”, “++X”, “--X”}
- b) operations = {“X++”, “++X”, “--X”, “X--”}

Sample input and output:

Input:

operations = {“++X”, “++X”, “X++”}

Output:

3

Explanation: The operations are performed as follows:

Initially, X = 0.

++X: X is incremented by 1, $X = 0 + 1 = 1$.

++X: X is incremented by 1, $X = 1 + 1 = 2$.

X++: X is incremented by 1, $X = 2 + 1 = 3$.

Question 2:

Given the array “nums” consisting of $2n$ elements in the form $[x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n]$.

Write a program to print the array in the form $[x_1, y_1, x_2, y_2, \dots, x_n, y_n]$ for the following two cases:

- a) $\text{nums} = \{1, 1, 2, 2\}$, $n = 2$
- b) $\text{nums} = \{1, 2, 3, 4, 4, 3, 2, 1\}$, $n = 4$

Sample Input and Output:

Input:

$\text{nums} = \{2, 5, 1, 3, 4, 7\}$, $n = 3$

Output:

$\{2, 3, 5, 4, 1, 7\}$

Explanation:

$n = 3$, hence, divide the input array into two arrays with three elements in each array.

$\{2, 5, 1\}$ $\{3, 4, 7\}$. This is of the form $\{x_1, x_2, x_3\}$ and $\{y_1, y_2, y_3\}$ respectively.

The output array must be of the form $\{x_1, y_1, x_2, y_2, x_3, y_3\}$. Therefore, the output array is $\{2, 3, 5, 4, 1, 7\}$.

Question 3:

A binary array is an array in which each element is either 0 or 1.

Given a binary array “nums”, write a program to print the maximum number of consecutive 1's in the array for the following two cases:

a) nums = {1,0,1,1,0,1}

b) nums = {1, 0, 0, 1}

Sample Input and Output:

Input:

nums = {1,1,0,1,1,1}

Output:

3

Explanation:

The first two digits or the last three digits are consecutive 1s. The maximum number of consecutive 1s is 3.

Question 4:

A subarray is a contiguous non-empty sequence of elements within an array.

Given an integer array `nums`, write a program to print the number of subarrays filled with 0 for the following two cases:

a) `nums = {1,3,0,0,2,0,0,4}`

b) `nums = {2,10,2019}`

Sample Input and Output:

Input:

`nums = {1,3,0,0,2,0,0,4}`

Output:

6

Explanation:

There are 4 occurrences of [0] as a subarray.

There are 2 occurrences of [0,0] as a subarray.

There is no occurrence of a subarray with a size more than 2 filled with 0.

Therefore, we return 6.

Linked List

Question 5:

The middle node of a linked list of size n is the $\lfloor n / 2 \rfloor$ th node from the start using 0-based indexing, where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x .

Write a program to create a Linked List and print the Linked List after deleting the middle node for the following two cases:

- a) 1->3->4->7->1->2->6->NULL
- b) 2->1->NULL

Sample Input and Output:

Input:

Linked List is 1 -> 2 -> 3 -> 4 -> NULL

Output: 1->2->4->NULL

Explanation:

Total number of nodes = 4.

Middle node is $\lfloor 4 / 2 \rfloor$ th node, which is node 2.

Here, the value of node with index 2 is 3.

Therefore, the Linked List becomes 1->2->4->NULL.

Stack

Question 6:

An input string is valid if:

- i) Open brackets must be closed by the same type of brackets.
- ii) Open brackets must be closed in the correct order.
- iii) Every close bracket has a corresponding open bracket of the same type.

Given a string "s" containing just the characters '(', ')', '{', '}', '[' and ']'. Write a program to determine if the input string is valid or not for the following two cases:

- a) s = "()"
- b) s = "([{}])"

Sample Input and Output:

Input:

s = "()"

Output: true

Explanation:

All the three conditions are satisfied by the string, hence it is a valid string.

Therefore, the output is true.

Queue

Question 7:

There are n people in a line queuing to buy tickets, where the 0th person is at the front of the line and the $(n - 1)$ th person is at the back of the line.

Each person takes exactly 1 second to buy a ticket. A person can only buy 1 ticket at a time and has to go back to the end of the line (which happens instantaneously) in order to buy more tickets. If a person does not have any tickets left to buy, the person will leave the line.

Given a 0-indexed integer array `tickets` of length n where the number of tickets that the i th person would like to buy is `tickets[i]`, Write a program to print the time taken for the person at position k (0-indexed) to finish buying tickets for the following case:

a) `tickets = {2,3,2}`, $k = 2$

Sample Input and Output:

Input:

`tickets = {5,1,1,1}`, $k = 0$

Output:

8

Explanation:

In the first pass, everyone in the line buys a ticket and the line becomes `{4, 0, 0, 0}`.

In the next 4 passes, only the person in position 0 is buying tickets.

The person at position 0 has successfully bought 5 tickets and it took $4 + 1 + 1 + 1 + 1 = 8$ seconds.

Tree

Question 8:

Write a program to create a Tree and traverse the Tree using Pre-Order for the following cases:

a) 15
1 14 3 7 4 5 15 6 13 10 11 2 12 8 9

Sample Input and Output:

Input:

6

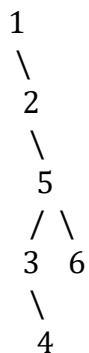
1 2 5 3 6 4

Output:

1 2 5 3 4 6

Explanation:

Input the tree



Output:

Pre-Order Traversal follows Root->Left Search -> Right Search
Therefore, the output will be 1 2 5 3 4 6.

AVL TREE

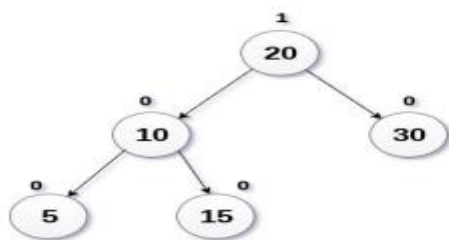
Question 9:

Construct AVL Tree for the following sequence of numbers-

50 , 20 , 60 , 10 , 8 , 15 , 32 , 46 , 11 , 48

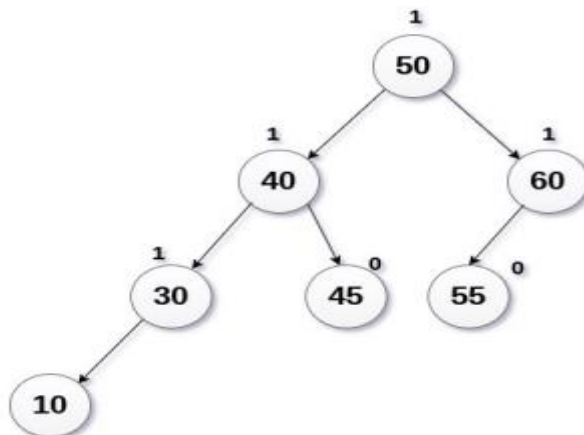
Question 10 a:

Delete the node 30 from the AVL tree shown in the following image.



Question 10 b:

Delete Node 55 from the AVL tree shown in the following image.



AVL Tree