

# Trilogy Programming Interview Questions

**Q1. Write a function to reverse a linked list in  $O(N)$  time and  $O(1)$  space.**

- a) Traverse list using 3 pointers and reverse links carefully.
- b) Repetitively pop a node from head of the list and insert into a new list at head position

**Q2. Given an array of  $n$  integers and an element  $k$ . Find whether there exists any two integers  $x, y$  whose sum is  $k$ .**

a) Sort and search takes  $O(n \log n)$  time and  $O(1)$  space.

- i. Sort the array in  $O(n \log n)$  using Heapsort.*
- ii. Maintain two pointers  $i, j$  at both ends of list.*
- iii.  $i=0; j=n-1$*
- iv. while( $i < j$ )*
  - 1. If  $a[i]+a[j]>k$  then  $j--$*
  - 2. else if  $a[i]+a[j]<k$  then  $i++$*
  - 3. else return true*
- v. }*
- vi. Return false;*

b) If the numbers are non-negative, use  $O(k)$  space i.e., a  $k+1$  bit array for getting an  $O(n)$  time solution, that too in a single parse.

**Q3. Given an array of positive integers. All numbers occur even number of times except 1 number which occur odd number of times. Find the number in  $O(n)$  time & constant space.**

a) Simply XOR all the elements in  $x$ . At the end  $x$  will have that element.

```
for (i = 1 to n) {  
    num = num ^ array[i];  
    return num;}  

```

# Trilogy Programming Interview Questions

**Q4. Convert binary tree to link list, (level by level bft) without copying the node, u can use one of the left or right pointer(decide which one) as next pointer to link list. Print binary tree level by level, but from last level to first.**

a) Use Queue for bft, but enter right child first & then left child. & on popping from queue put it on stack. At last pop stack & print.

**Q5. Write a function to traverse a tree in inorder/ preorder/ postorder.**

a) Recursive routine {O(N) time and O(log n) space}

b) Recursive routine without tail recursion

c) Iterative solution using Stack

**Q6. Given an array of n elements. Elements of array are number between 1 to n. only 1 number in which is occurring twice and only 1 is not present. Find which number is repeating (r) and which is missing (m).**

a) Can be done in O(1) time.

i. Find sum of numbers and subtract it from  $n(n-1)/2$ . You get (r-m)

ii. Find sum of squares of numbers and subtract it from  $n^2(n-1)^2/2$ . You get (r<sup>2</sup>-m<sup>2</sup>)

iii. You have two equations and two unknowns. Solve them.

**Q7. We have two strings A and B with the same super set of characters. We need to change these strings to obtain two equal strings. In each move we can perform one of the following operations:**

1. swap two consecutive characters of a string

2. swap the first and the last characters of a string

A move can be performed on either string.

What is the minimum number of moves that we need in order to obtain two equal strings?

**Q8. Write a code in which the client registers some method on Registrar and the registrar calls that method in case some event happens.**

# Trilogy Programming Interview Questions

**Q9. Write an algorithm to divide two extremely large numbers, which cannot be stored in an int, long int, float, double etc. Find the remainder and quotient.**

Notes:

1. Do not use the subtraction approach.

**Q10. Create a strategy for testing flipkart.com (any e-com site)**

Suppose it is under development.

- What all positive scenarios to be tested? Negative scenarios to be tested?
- How you will ensure integration?
- What all type of testing should be done?