

DSA TEST
Lakshit Verma
Sec – A
12218002722
IV Year

Q1- Reverse a Linked List

Solution:

```
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

def reverseList(head):
    prev = None
    curr = head
    while curr:
        next_node = curr.next
        curr.next = prev
        prev = curr
        curr = next_node
    return prev
```

Q2- Is Subtree?

Solution:

```
class TreeNode:
```

```
    def __init__(self, val=0, left=None, right=None):
```

```
        self.val = val
```

```
        self.left = left
```

```
        self.right = right
```

```
def isSubtree(root, subRoot):
```

```
    if not root:
```

```
        return False
```

```
    if isSameTree(root, subRoot):
```

```
        return True
```

```
    return isSubtree(root.left, subRoot) or isSubtree(root.right, subRoot)
```

```
def isSameTree(p, q):
```

```
    if not p and not q:
```

```
        return True
```

```
    if not p or not q:
```

```
        return False
```

```
    if p.val != q.val:
```

```
        return False
```

```
    return isSameTree(p.left, q.left) and isSameTree(p.right, q.right)
```