

LeetCode Bootcamp Week 6 Submission
Submitted By: Raunak Choudhary (NetID: rc5553)

Question1: Lowest Common Ancestor of a Binary Tree

Code:

```
class Solution(object):
    def lowestCommonAncestor(self, root, p, q):
        if not root or root == p or root == q:
            return root

        left = self.lowestCommonAncestor(root.left, p, q)
        right = self.lowestCommonAncestor(root.right, p, q)

        if left and right:
            return root
```

Question2: Top K Frequent Elements

Code:

```
class Solution(object):
    def topKFrequent(self, nums, k):
        count = {}
        for num in nums:
            count[num] = count.get(num, 0) + 1

        bucket = [[] for _ in range(len(nums) + 1)]
        for num, freq in count.items():
            bucket[freq].append(num)

        result = []
        for i in range(len(bucket) - 1, 0, -1):
            if bucket[i]:
                result.extend(bucket[i])
                if len(result) >= k:
                    return result[:k]

        return result
```

Question 3: Word Break

Unable to solve this problem due to not having a LeetCode Premium Account.