LeetCode Bootcamp Week 8 Submission
Submitted By: Raunak Choudhary (NetID: rc5553)

Question 1: Partition Equal Subset Sum

Code:

```
class Solution(object):
    def canPartition(self, nums):
        total_sum = sum(nums)
        if total_sum % 2 != 0:
            return False
        target = total_sum // 2

        dp = [False] * (target + 1)
        dp[0] = True

        for num in nums:
            for i in range(target, num - 1, -1):
                dp[i] = dp[i] or dp[i - num]

        return dp[target]
```

Question 2: Coin Change

Code:

```
class Solution(object):
    def coinChange(self, coins, amount):
        dp = [amount + 1] * (amount + 1)
        dp[0] = 0

        for coin in coins:
            for i in range(coin, amount + 1):
                dp[i] = min(dp[i], dp[i - coin] + 1)

        return dp[amount] if dp[amount] != amount + 1 else -1
```

Question 3: Maximum Subarray

Code:

```
class Solution(object):
    def maxSubArray(self, nums):
        current_sum = nums[0]
        max_sum = nums[0]

        for i in range(1, len(nums)):
            current_sum = max(nums[i], current_sum + nums[i])

            max_sum = max(max_sum, current_sum)

        return max_sum
```