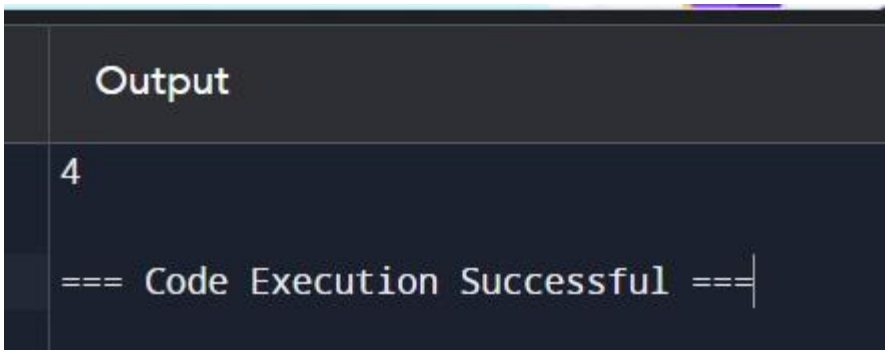
195. You are given a 0-indexed integer array coins, representing the values of the coins available, and an integer target. An integer x is obtainable if there exists a subsequence of coins that sums to x. Return the minimum number of coins of any value that need to be added to the array so that every integer in the range [1, target] is obtainable. A subsequence of an array is a new non-empty array that is formed from the original array by deleting some (possibly none) of the elements without disturbing the relative positions of the remaining elements.

Program:

```
def minCoinsToAdd(coins, target):
    coins.sort()
    max_reachable = 0
    coins_needed = 0
    for coin in coins:
        if coin > max_reachable + 1:
            break
        max_reachable += coin
    while max_reachable < target:
        max_reachable += max_reachable + 1
        coins_needed += 1
    return coins_needed
# Example
coins = [1, 4, 10]
target = 19
print(minCoinsToAdd(coins, target))
```

Output:



Time complexity: O(nlogn