

1. 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.

Example 1:

Input: `coins = [1,4,10]`, `target = 19`

Output: 2

Explanation: We need to add coins 2 and 8. The resulting array will be `[1, 2, 4, 8, 10]`. It can be shown that all integers from 1 to 19 are obtainable from the resulting array, and that 2 is the minimum number of coins that need to be added to the array.

AIM: To write a python program for the . 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 min_coins_to_target(coins, target):
    coins.sort()
    current_sum = 0
    added_coins = 0
    i = 0

    while current_sum < target:
        if i < len(coins) and coins[i] <= current_sum + 1:
            current_sum += coins[i]
            i += 1
        else:
            added_coins += 1
            current_sum += current_sum + 1

    return added_coins

coins = [1, 4, 10]
target = 19
print(min_coins_to_target(coins, target))
```

output: 

TIME COMPLEXITY : $O(n \log n)$