Logo

STUDENT REPORT

DETAILS

Name

MADHU.K

Roll Number

3BR23EE057

EXPERIMENT

Title

OBJECT SCORE

Description

In a family, there are N members each have a capacity of Ci units to buy anything. In a store there are M objects. Each of which have some price Pi and weight Wi print on it. Each of the members go to the store and can buy all those items whose price is less than or equal to their buying capacity and store that bought object in a bag. Find the maximum weight of each of the bags collected by all N members individually.

Input Format:
First line contains two integers N and M where N is the number of members in the house and M is the number of objects in the store.
Second line contains N space-separated integers (C1, C2, C3,...)

the next M lines contains each object price and weight(Pi,Wi) as space seperated integers.

Sample Input:

3 4

10 20 30

5 10

15 20

10 25

20 30

Sample Output:
35 85 85

Source Code:

```
def max_weight_for_capacity(objects, capacity):
    # Create a DP array to store the maximum weight for each capacity
    dp = [0] * (capacity + 1)

    for price, weight in objects:
        for c in range(capacity, price - 1, -1):
            dp[c] = max(dp[c], dp[c - price] + weight)

    return dp[capacity]

def main():
    # Read input values
    n, m = map(int, input().strip().split())
    capacities = list(map(int, input().strip().split()))

    objects = []
    for _ in range(m):
        p, w = map(int, input().strip().split())
        objects.append((p, w))

    # Find maximum weight for each member based on their capacity
    results = []
    for capacity in capacities:
        max_weight = max_weight_for_capacity(objects, capacity)
        results.append(max_weight)

    # Print results for all members
    print(" ".join(map(str, results)))

# Sample Input
main()
```

RESULT

0 / 5 Test Cases Passed | 0 %