Dynamic Programming:

Raju is having a gold biscuit of length n and a list of prices of gold biscuit of length i where 1<=i<=n, find the optimal way to cut the gold biscuit into smaller biscuits in order to maximize profit.

Example:

Consider below gold biscuit lengths and values.

Sample Input:

length[]=[1,2,3,4,5,6,7,8]
price[]=[1,5,8,9,10,17,17,20]

Biscuit length:4

Best Solution:

Cut The biscuit in to two pieces of length 2 each to gain revenue of 5+5=10

Cut	**Profit**
4	9
1,3	1+8=9
2,2	5+5=10
3,1	8+1=9
1,1,2	1+1+5=7
1,2,1	1+5+1=7
2,1,1	5+1+1=7
1,1,1,1	1+1+1+1=4

Sample Output:

10

Solution:

```
def biscuit(price,n):
    if n==0:
        return 0
    maxValue=float('-inf')
    for i in range(1,n+1):
```

```
        cost=price[i-1]+biscuit(price,n-i)
        if cost>maxValue:
            maxValue=cost
    return maxValue
price=list(map(int,input().split()))
n=int(input())
print(biscuit(price,n))
```