

# 2294번: 동전 2

[2294번](#)[제출](#)[맞은 사람](#)[숏코딩](#)[풀이](#)[풀이 작성](#)[풀이 요청](#)[재채점/수정](#)[문제 추천](#)[채점 현황](#)[내 소스](#)[강의▼](#)[질문 검색](#)[질문 작성](#)

## 동전 2

[성공](#)[풀이](#)

시간 제한

메모리 제한

제출

정답

맞은 사람

정답 비율

2 초

128 MB

15385

3975

2740

25.689%

## 문제

n가지 종류의 동전이 있다. 각각의 동전이 나타내는 가치는 다르다. 이 동전들을 적당히 사용해서, 그 가치의 합이 k원이 되도록 하고 싶다. 그러면서 동전의 개수가 최소가 되도록 하려고 한다. (각각의 동전은 몇개라도 사용할 수 있다.)

```
n ,k = map(int,input().split())
coins = [ int(input()) for i in range (n)]

dp = [0 for i in range (100001) ]
# dp[0] =1
for i in coins : dp[i] = 1
for i in range (1,k+1):
    for j in coins :
        if i > j and dp[i-j] :
            cnd = dp[i-j]+1
            if not dp[i] : dp[i] = cnd
            else : dp[i] = cnd if cnd<dp[i] else dp[i]

print(dp[k] if dp[k] else -1)
```

생각 보다 쉬운 dp 였다.

$f(n)$  = n원을 만들 수 있는 최소 동전의 종류

$f(n) = 1$  if coins.\_\_contains\_\_(n)

$f(n) = \min(f(n-c1)+1, f(n-c2)+1, \dots)$

## 입력

첫째줄에  $n, k$ 가 주어진다. ( $1 \leq n \leq 100, 1 \leq k \leq 10,000$ ) 다음  $n$ 개의 줄에는 각각의 동전의 가치가 주어진다.

## 출력

첫째줄에 사용한 동전의 최소 개수를 출력한다. 불가능한 경우에는 -1을 출력한다.

## 예제 입력

```
3 15
1
5
12
```

## 예제 출력

```
3
```

## 힌트

## 출처

- 잘못된 조건을 찾은 사람: apples1309
- 데이터를 추가한 사람: isac322

## 알고리즘 분류

- 다이나믹 프로그래밍
- 동전 교환

