

# Bài toán cái túi - version I

Khái niệm chung về quy hoạch động (Dynamic programming)

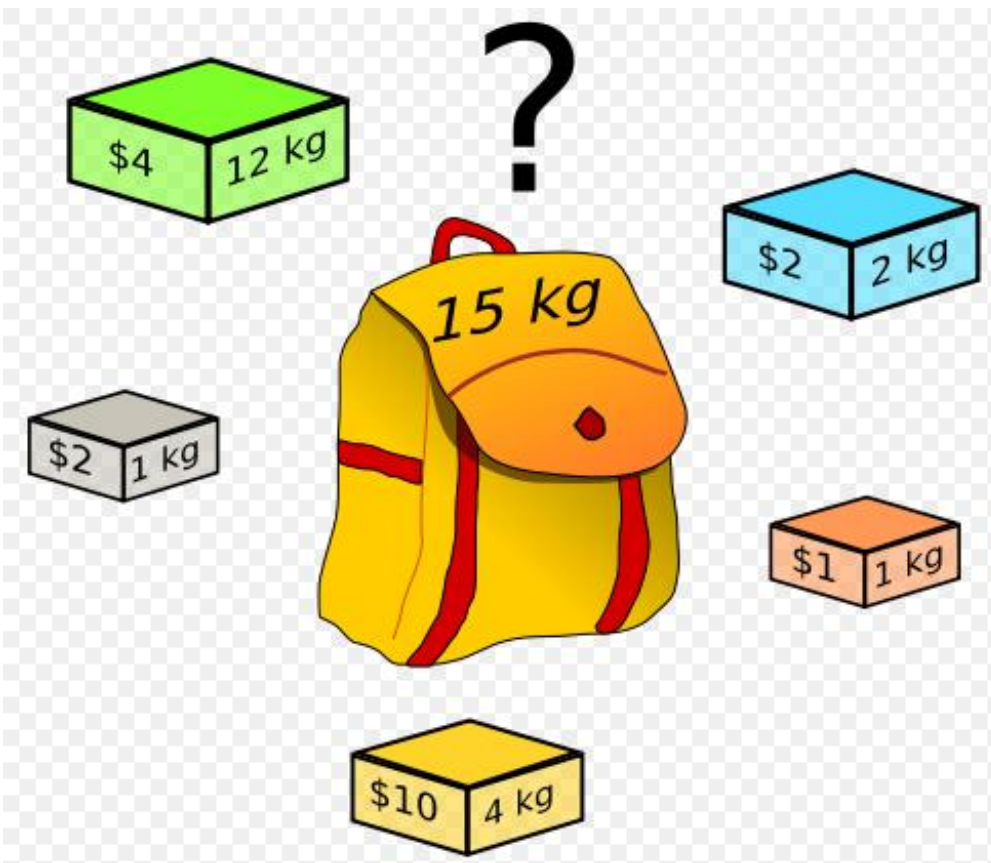
Ta có thể giải một bài toán với cấu trúc con tối ưu bằng một quy trình ba bước:

1. Chia bài toán thành các bài toán con nhỏ hơn.
2. Giải các bài toán này một cách tối ưu bằng cách sử dụng đệ quy quy trình ba bước này.
3. Sử dụng các kết quả tối ưu đó để xây dựng một lời giải tối ưu cho bài toán ban đầu.

Các bài toán con được giải bằng cách chia chúng thành các bài toán nhỏ hơn, và cứ tiếp tục như thế, cho đến khi ta đến được trường hợp đơn giản để tìm lời giải.

Bài toán xếp balô (knapsack problem)

Bài toán được đặt tên từ vấn đề chọn những gì quan trọng có thể nhét vừa vào trong một cái túi (với giới hạn khối lượng) để mang theo trong một chuyến đi.



Cho trước một tập gồm  $n$  đồ vật, mỗi đồ vật có một chi phí  $V_i$  và một giá trị  $v_i$ , xác định xem cần chọn những đồ vật nào sao cho tổng chi phí nhỏ hơn một ngưỡng cho trước (giới hạn của balô) và tổng giá trị cao nhất có thể được.

## Input

Dòng 1:  $n, S$  (với  $S$  là giới hạn của balô)

n dòng tiếp theo: Mỗi dòng gồm 2 số  $V_i$ ,  $W_i$  là chi phí và giá trị đồ vật thứ  $i$ .

## Output

Dòng 1: Ghi 2 số nguyên  $\$, k$  là tổng giá trị lớn nhất tìm được và số đồ vật được chọn

k dòng tiếp theo ghi chỉ số các đồ vật được chọn

## Example

### Input:

```
5 15
12 4
2 2
1 1
1 2
4 10
```

### Output:

```
15 4
2
3
4
5
```

### Giới hạn:

```
1 <= n <= 1000
1 <= m <= 1000
1 <=  $W_i$  <=  $10^5$ 
1 <=  $V_i$  <=  $10^4$ 
```