

Naive Knapsack

Description

This is the classical multiple knapsack problem. There are N types of items and a knapsack with a capacity of C , where the quantity of each item is limited. The i -th item has a cost of $v[i]$, a value of $w[i]$, and a quantity of $s[i]$.

The question is which items to choose and how many of each item to choose in order to maximize the total value.

Input Format

The first line includes two integers: N, C .

For the next N lines, each containing three integers $v[i], w[i], s[i]$.

Output Format

You only need to print one integer, representing the maximum total value you can get.

Sample

Sample Input

```
2 5
1 1 2
2 2 1
```

Sample Output

```
4
```

For 40% testcases: $1 \leq N, C, v[i], w[i], s[i] \leq 100$.

For 80% testcases: $1 \leq N, C, v[i], w[i], s[i] \leq 1000$.

For 100% testcases: $1 \leq N, C, v[i], w[i], s[i] \leq 4000$.

Notes

You can use brute-force solution with $O(N^3)$ time to pass the first 40% testcases.

You can use the optimized solution with $O(N^2 \log N)$ time to pass 80% testcases.

If you want to pass all testcases, make sure that your solution has a tight $O(N^2)$ time.

