#Problem 1: Simple Coin Change

Description:
You will be given N coins and an amount K. You can use each coin an infinite number of times. You have to calculate the minimum number of coins needed to make the amount K.

In the first line you will be given N and K. In the following line, you will be given N values $C_1,C_2,C_3,...,C_N$ denoting N coins.

For each set of input, there will be a single line of output denoting the minimum number of coins to make K.

Limits
1<=N<=100000
1<=k<=1000000
Each coin C_i will have values between 1 and 10000 inclusive

Test Cases:

Input	Output
8 6 1 2 5 10 20 50 100 200	2
8 170 1 2 5 10 20 50 100 200	3
8 2 1 2 5 10 20 50 100 200	1
8 201 1 2 5 10 20 50 100 200	2
8 217 1 2 5 10 20 50 100 200	4

#Problem 2: Fractional Knapsack

Description:
You will be given N items and a knapsack weight W. Each item N[i] has two elements, its positive benefit per unit b[i] and its total weight w[i]. You have to choose the elements in such a way that you can maximize your total benefit. You can take a fractional amount of weights for each item.

In the first line, you will be given N and W. In the first following line you will be given N values, each element denotes the positive benefit per unit where ith value is for the ith element. In the second following line, you will again be given N values, each denoting the weights where ith value denotes the total available weight for the ith element.

For each set of input, there will be a single line of output denoting the optimal value that maximize your total benefit.

Limits
1<=N<=100000
1<=W<=100000
1<=b[i]<=100
1<=w[i]<=100000

Test Cases:

Input	Output

5 10
3 4 20 5 50
4 8 2 6 1
	124

5 12.5
3 4 20 5 50
4 8 2 6 1
	134

5 7
3 4 20 5 50
4 8 2 6 1
	110

#Problem 3: Activity Scheduling

Description:
You will be given N tasks, each having a starting and ending time. You can complete only one task at a time. You have to schedule the tasks in such an order so that the number of completed tasks maximizes.

In the first line, you will be given a value N. In the following N lines, you will be given the information of each N tasks where the ith line contains the starting time S_i and ending time E_i for the ith task.

For each set of input, there will be a single line of output denoting the number of maximum possible tasks that can be performed without violating conflicting constraint. Overlapping between consecutive tasks' ending and starting time is permissible, e.g., one task ended at 3 and another task started at 3, in this case, both tasks can be considered without considering the overlapping conflict.

Limits
1<=N<=100000
1<=S_i<=1000
1<=E_i<=1000 always E_i >=S_i for all i

Test Cases:

Input	Output

4
1 10
2 3
4 7
8 12
	3

3
1 2
2 3
4 5
	3

4
1 2
3 4
5 6
7 8
	4

3
1 10
10 11
11 20
	3