The Kebab eating contest (kebab)

In the city of Sarajevo, in Bash-Carsija, a kebab eating contest is held. There are *n* restaurants that sell kebab sandwiches (containing different number of kebabs inside), and *m* contestants. In every restaurant the kebab sandwich has the same price of 1 KM.

Kebab restaurants are so well trained and equipped with huge grills that they can make and serve kebab sandwiches to any number of customers almost instantly. They are also very well stocked, so they never run out of kebabs. To increase efficiency, each restaurant makes only one kind of kebab sandwich with the same number of kebabs inside.

The point of this contest is to eat the largest number of kebabs (individual kebabs, not sandwiches).

During the eating contest, restaurants keep track of the contestants and sell at most one sandwich to each contestant. Restaurants open at different times but they all close at the end of the contest. The opening time for restaurant *i* is specified in *open[i]*. Number of kebabs in the *i*-th restaurant's sandwich is specified with *size[i]*.

The *j*-th contestant is given time unit *time[j]* when he can buy sandwiches and *km[j]* amount of money. Each contestant needs just one time unit to visit all the restaurants he can to maximize the number of his kebabs.

Contestant can visit a restaurant only if it has opened before or at the time contestant's given time (*open[i]<=time[j]).*

Task

Your task is to write procedure *KebabCount* that takes seven parameters. The first one is *n* – the number of restaurants. The second and third parameters are arrays of *n* integers representing the opening times of restaurants and the number of kebabs in each restaurant's sandwich, respectively. The fourth parameter is *m* – number of contestants. The fifth and sixth parameters are arrays of *m* integers representing contestant's time units and amount of money given to each of the contestants, respectively. You should return your results in

the seventh parameter which is an array of *m* integers representing the maximum number of kebabs each contestant can eat.

Example

KebabCount(2, (4, 4), (5, 2), 4, (5, 4, 4, 2), (2, 1, 2, 1), count); count = (7,5,7,0)

We have two restaurants with opening times at 4 time units from the start and make sandwiches that have 5 and 2 kebabs. There are four contestants which buy sandwiches at 5, 4, 4 and 2 time units from the start and have 2, 1, 2 and 1 KM. Contestants can eat a maximum of 7, 5, 7 and 0 kebabs.

Subtasks

Subtask 1 (4 points) : 1<=*m,n*<=10; 1<=*open[i],time[j]*<=1,000; 1<=*size[i]*<=1,000; 1<=*km[j]*<=n

Subtask 2 (12 points) : 1<=*m,n*<=100; 1<=*open[i],time[j]*<=2,000,000,000; 1<=*size[i]*<=20,000; 1<=*km[j]*<=n

Subtask 3 (21 points) : 50<=*m,n*<=100,000; 1<=*open[i],time[j]*<=2,000,000,000; 1<=*size[i]*<=50; 1<=*km[j]*<=50

Subtask 4 (26 points) : 1<=*m,n*<=100,000; 1<=*open[i],time[j]*<=2,000,000,000; 1<=*size[i]*<=20,000; 1<=*km[j]*<=n; km[k] = km[l] for all k, l.

Subtask 5 (37 points) : 1<=*m,n*<=100,000; 1<=*open[i],time[j]*<=2,000,000,000; 1<=*size[i]*<=20,000; 1<=*km[j]*<=n