Team Arrangement

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	512 megabytes

Misconception of the majority of teachers: w_i is a strictly increasing sequence.

— Master Chicken, colleague of Master Bo

There are n students attending the algorithm class held by Master Bo. Bo asks the students to do teamwork, and then helps them divide into teams.

Each student must belong to exactly one of the teams. Master Bo knows his students well and knows that the i-th will be satisfied with the division only if the number of students in his team is not less than l_i and not greater than r_i (including himself). Note that a team can consist of exactly one student.

You will be given n integers $w_1, w_2, \ldots, w_n$. Assume there will be m teams finally, the i-th of which consists of c_i students, the **weight** of such an arrangement will be $w_{c_1} + w_{c_2} + \cdots + w_{c_m}$.

Master Bo is now wondering how to divide students into teams such that every student will be satisfied, and the **weight** of the arrangement is maximized. Please write a program to help Master Bo.

Input

The first line of the input contains a single integer n ($1 \leq n \leq 60$), denoting the number of students.

In the next n lines, the i-th line contains two integers l_i and r_i ($1 \leq l_i \leq r_i \leq n$), describing the i-th student.

The next line contains n integers $w_1, w_2, \ldots, w_n$ ($|w_i| \leq 10^7$).

Output

Print a single line containing an integer: the maximum possible value of the weight. If it is impossible to find such an arrangement, please print "`impossible`" instead.

Examples

standard input	standard output
3 2 3 1 2 2 2 4 5 100	9
3 1 3 3 3 2 3 1 1 100	100
2 1 1 2 2 1 1	impossible
3 2 3 1 2 2 2 -100 -200 100000	-300

Note

The last example illustrates why ICPC requires teams of three.