

2018 Canadian Computing Olympiad

Day 1, Problem 3

Fun Palace

Time Limit: 1 second

Problem Description

You are working hard to prepare a fun party for your fun friends. Fortunately, you have just located the perfect venue for the fun party: a *fun palace*. The fun palace has N *fun rooms* connected in a linear structure. The fun rooms are numbered from 1 to N , and for $1 \leq i \leq N - 1$, fun rooms i and $i + 1$ are connected by a *fun tunnel*. We say that such a fun tunnel is *incident* to fun rooms i and $i + 1$. In addition, fun room 1 is incident to an *exit tunnel* leaving the fun palace.

Fun tunnels can be in one of two states: open or closed. When the fun tunnel between fun rooms i and $i + 1$ is opened, fun friends may travel freely between the two rooms, in either direction.

By default, the fun tunnels will all be closed. However, they may temporarily be opened by a group of fun friends pressing down a required number of *fun buttons*. For each fun tunnel, there is a set of fun buttons present in the fun rooms connected to the fun tunnel. If all of the buttons in one of the rooms connected to a tunnel are pressed down by distinct fun friends, then the fun tunnel will open. Otherwise, the fun tunnel will immediately close. The fun tunnel between rooms i and $i + 1$ is connected to a set of a_i buttons in room i and a set of b_i buttons in room $i + 1$. To put this another way, if there are at least a_i friends in room i or if there are b_i friends in room $i + 1$, then tunnel between room i and $i + 1$ may be opened.

The exit tunnel operates under similar rules, but it is only connected to a single set of e buttons present in room 1.

You want to ensure your friends have maximum fun, and that obviously means keeping your fun friends trapped in the fun palace forever. What is the maximum number of fun friends that you can distribute to particular fun rooms such that the exit fun tunnel is never opened?

Input Specification

The first line will contain a single integer N ($1 \leq N \leq 1000$), the number of fun rooms. The next line contains a single integer e ($1 \leq e \leq 10\,000$). The next $N - 1$ lines contain two space-separated integers each, with the i th of these lines containing a_i and b_i ($1 \leq a_i, b_i \leq 10\,000$).

For 3 of the 25 marks available, $1 \leq e \leq 200$, $a_i = 1$, $b_i = 1$.

For an additional 5 of the 25 marks available, $1 \leq e, a_i, b_i \leq 2$.

For an additional 12 of the 25 marks available, $N \leq 200$, $1 \leq e, a_i, b_i \leq 200$.

Output Specification

Output a single integer, the maximum number of fun friends over all possible distributions of fun friends to fun rooms such that there is no way for the fun friends to open the exit tunnel.

Sample Input 1

2
20
5 5

Output for Sample Input 1

19

Explanation for Output for Sample Input 1

If we had any more than 19 fun friends, then they would be able to move to fun room 1 and press all 20 fun buttons required in order to open the exit tunnel.

Sample Input 2

2
20
5 20

Output for Sample Input 2

24

Explanation for Output for Sample Input 2

Suppose we place 24 fun friends in fun room 2. In order to open the fun tunnel between fun rooms 1 and 2, 20 fun friends must stay in fun room 2 to press the fun buttons. This allows only 4 fun friends into fun room 1, which is not enough to press every fun button in the set of 5.

Sample Input 3

7
7
2 8
6 6
1 1
2 4
2 8
7 8

Output for Sample Input 3

23

Explanation for Output for Sample Input 3

One optimum distribution is to place 9 fun friends in fun room 2 and 14 fun friends in fun room 7.