Problem D. Biochips

Input file:	`d.in`
Output file:	`d.out`
Time limit:	2 seconds
Memory limit:	512 megabytes

Scientists discovered a biochip, which can divide itself into several new biochips. Herewith the parent-biochip disappears. Each biochip has its own size of memory, the size does not depend on the parent memory size. Then the biochip might be either used (stopping its division), or it will continue division in a similar manner.

Scientists prepared tree-like schemes of biochip-division process and they know the exact structure of the tree consisting of N elements, including the memory size of each of possible biochips descendants.

Make a program to choose from the tree exactly M biochips total memory size of which is the biggest possible. Pay attention to the fact that when we are choosing a biochip, we can't choose neither any of its ancestors nor descendants.

Input

The first line of input file contains two integers N and M — number of elements of the tree and number of biochips to be chosen ($1 \leq N \leq 200\,000$, $1 \leq M \leq 500$).

The following N lines contain two non-negative integers each: number of the parent in the tree and the size of the chip's memory x ($1 \leq x \leq 1000$). Each biochip has a unique number ranging from 1 to N inclusively. The line with i number includes information about a biochip with number $i - 1$. Just one biochip doesn't have a parent and it's parent is written as 0.

It's guaranteed that it's possible to choose M biochips.

Output

Output file consists of one integer — maximal possible amount of memory of M chosen biochips.

Examples

d.in	d.out
7 3	300
2 100	
0 1000	
2 150	
3 100	
3 5	
5 100	
2 50	

Note

In 20% of testcases $N \leq 20, M \leq 10$.

In 60% of testcases $N \leq 10\,000, M \leq 100$.