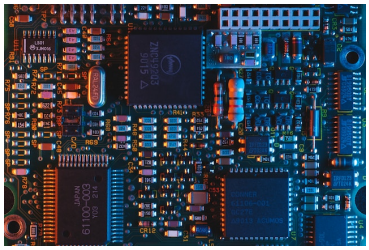
Problem E
Electronic Components



Picture by Umberto, Unsplash Licence

Sara is doing her summer internship at NCPC (Never Crashing Personal Computers). One day, a rare creature appeared in the office: an algorithmic problem!

The company has a machine that places electronic components on circuit boards. Normally, it would do this one component at a time. But recently the machine has received an update which allows it to place two different components simultaneously. The bottleneck then becomes the component with greater placement time. Now it is far from obvious what strategy the machine should use in order to minimize the total placement time. Sara decides to write an algorithm to determine this strategy.

You have N different types of electronic components. There are f_i copies of the ith type, and the components of this type have a placement time of t_i nanoseconds. The goal is to place all of the components using a sequence of moves. In one move, the machine can take two components of type i and j, where $i \neq j$, and place both of them simultaneously. This takes $\max(t_i, t_j)$ nanoseconds. The machine can also place a single component of type i in one move, which takes t_i nanoseconds.

Calculate the minimum possible time to place all components.

Input

The first line of input contains the integer N ($1 \leq N \leq 1000$).

The following N lines each contain two integers f_i and t_i ($1 \leq f_i \leq 10^4$, $1 \leq t_i \leq 10^9$).

Output

Print one integer, the minimum time to place all components.

Sample Input 1	Sample Output 1
3 2 7 2 1 3 10	31

Sample Input 2	Sample Output 2
3 2 10 2 11 2 12	35

Sample Input 3	Sample Output 3
4 2 11 7 10 3 5 1 1	72