Problem D
Fail Fast
Time Limit: 4 Seconds, Memory Limit: 2G

A large software project has been written with many automated tests to help ensure quality. When a change is made to the source code, all of the automated tests are run in some sequence to help show that the code change has not broken some pre-existing functionality. The code change is accepted if and only if all automated tests pass.

Running all of these automated tests is costly, so as soon as there is one failure, the remaining tests are not run. Each test takes a certain amount of CPU time to perform. The cost of running a sequence of tests when there is a test failure is the sum of the CPU time used by the automated tests up to and including the failing test. The cost of running a sequence of tests where all tests pass is equal to 0.

A given automated test may rely on the output of another single automated test. For example, automated test **A** may depend on the output of automated test **B**. In such a case, test **B** must be executed before test **A**. Note that other tests may be run between test **B** and test **A**.

Based on an analysis of past data and an assessment of the code change, each automated test has a known probability of passing. The probability of any given test passing is independent of the probability of any other test passing.

Given the CPU time to execute each test, the probability of each test's failure, and the dependencies, determine an order in which the tests should be executed, sequentially, to minimize the expected cost of running the sequence of tests.

Input

The first line of input contains a single integer n ($1 \leq n \leq 10^5$), which is the number of automated tests.

Each of the next n lines contains three values, integer c ($1 \leq c \leq 10^6$), real number p ($0 < p < 1$), and integer d ($0 \leq d \leq n$), where c is the CPU time needed to run the automated test, p is the probability of the test passing, and d is the index of the test on which this one depends or 0 if the test has no dependency. The tests are indexed from 1 to n in the order in which they are given in the input.

Each probability is specified with at most six decimal digits. There are no cyclic dependencies.

Output

Output n lines, each line containing a single integer, giving the indices of the tests to run in sequence. The ordering must minimize the expected cost of running the tests to within 10^{-6} absolute or relative error of the expected cost of the optimal order. Any ordering which satisfies this constraint will be accepted.

Sample Input 1	Sample Output 1
4 100 0.5 0 200 0.1 1 10 0.5 2 10 0.9 0	4 1 2 3