For the analysis of vehicle traffic on a one-way street, on one lane, the road signaling company has installed a system that automatically detects the vehicles that pass on the street. On the street pass only vans and small vehicles. The system is formed by two presence sensors placed at 1 meter spacing, along the street, at 50 cm above the ground. The sensors are sampled at a frequency which is high enough and the samples are sent to the analysis system. A sample is formed by the time stamp (a number that represents the number of milliseconds from the start of the system), and the state of the sensor: 1 if it detects an object in front of it and 0 if not.

Requirement

Given a positive number **n** of samples, then **n** samples for the left sensor and then **n** samples for the right sensor, one should compute the average speed of the vehicles (computed as the arithmetic mean of speed of all the vehicles and also the number of vehicles).

Input data

On the first line is positive integer **n**, that represents the number of samples, followed by the *newline* character. On the next **2 * n** lines are the samples in the format "**time_stamp sensor_state**", the first **n** samples being for the left sensor and the next **n** samples being for the right sensor.

Output data

Display, on a single line, two integer values: the average speed of the vehicles, in kilometers per hour, computed by rounding and the number of vehicles, the values being separated by a blank space.

ATTENTION to the compliance to the problem requirements: the display of results must be done EXACTLY as required! In other words, on the standard output stream there will be nothing displayed in addition to the problem requirements; following the automatic evaluation, any supplemental character displayed, or any display different than the requirements, will produce an eroneous result and will lead to the „Reject" of the solution.

Restrictions and remarks

1. 3 < n <= 1000
2. The time stamp is a positive, 32 de bits integer.
3. The speed of a vehicle is constant in all the duration of its passage in front of the sensors.
4. Use float point double precision variables for storing fractional values.
5. Computation will be performed at maximal precision and rounding will be performed only for display.
6. The vehicles can have different lengths.
7. **Caution**: Depending on the programming language you have chosen, the file containing the code must have one of the .c, .cpp, .java, or .m extensions. The web editor will not automatically add these extensions and their lack leads to the impossibility of compiling the program!
8. **Caution**: for those working in MATLAB, it is recommended that the source file be named by the candidate in the form of: <name>.m where *name* is the candidate's surname.
9. **Caution**: There may be spaces or other white characters at the beginning or end of any input lines. The reading of the data must be carried out with this in mind!

Examples

Input	Output	Explanations
5 10000 0 10050 1 10100 1 10150 0 10200 0 10000 0 10050 0 10100 1 10150 1 10200 0	72 1	Each sensor has exactly one areas of 1 reading, so each has detected one vehicle. Using the timestamps, we can compute the speed at 72 kilometers/ hour.
11 54300 1 54400 1 54500 0 54600 1 54700 1 54800 0 54900 0 55000 1 55100 1 55200 0 55300 0 54300 0 54400 1 54500 1 54600 0 54700 1 54800 1 54900 0 55000 0 55100 1 55200 1 55300 0	36 3	Each sensor has three areas of 1 reading, so each has detected three vehicles. Using the timestamps, we can compute the speed at 36 kilometers/ hour.

Worktime: 150 minutes