

# Download Time Monitor

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:         1024 megabytes

You are currently downloading something from a cloud storage, but the download speed is very disappointing. You feel that only monitoring the download speed is not enough to relieve your anxiety, so you start calculating the download time.

The network bandwidth you are using is  $B$  MiBps, and now you have two large files to download. The first file starts downloading at  $t_1$  second and has a size of  $a_1$  MiB, while the second file starts downloading at  $t_2$  second and has a size of  $a_2$  MiB. Due to your use of the most advanced congestion control algorithm, when only one file is downloading at any given time, the download rate will remain constant at  $B$  MiBps. When two files are downloading, the download rates of both files will remain constant at  $\frac{B}{2}$  MiBps. When a file starts or finishes downloading, the download rates of all files will instantly adjust to the target value.

Now you want to know how long it will take for each of the two files to finish downloading.

## Input

The first line contains an integer  $T$  ( $1 \leq T \leq 10^5$ ), indicating the number of test cases.

For each test case, one line with five integers  $B, t_1, a_1, t_2, a_2$  ( $1 \leq B, a_1, a_2 \leq 10^5, 0 \leq t_1, t_2 \leq 10^5, t_1 \leq t_2$ ), as described in the statement.

## Output

For each set of data, output a line with two real numbers, representing the time it takes for the first and second files to finish downloading. Your answer will be considered correct only if the relative or absolute error between your answer and the correct answer does not exceed  $10^{-6}$ .

## Example

standard input	standard output
3	2.000000000 0.800000000
5 0 10 10 4	2.666666667 0.666666667
3 2 7 2 1	3.666666667 3.666666667
3 2 7 3 7	

## Note

Please note that reading in a large number of floating-point numbers will cause an I/O slowdown.