

Fractional Knapsack

time limit per test: 2 seconds (2 seconds for Python)
memory limit per test: 976.6 megabytes
input: standard input
output: standard output
points: 100 · problem code: fracsack

You are packing a bag with weight capacity C . There are n items, and the i -th item has value v_i and weight w_i . You do not have to take an entire item. You may take any fraction of an item, and both its weight and value scale proportionally. In particular, if you take weight x from item i (where $0 \leq x \leq w_i$), then you gain value $x \cdot v_i/w_i$. What is the maximum total value you can obtain?

Input

The first line contains two integers n and C , the number of items and the knapsack capacity.

The next n lines each contain two integers v_i and w_i , the value and weight of an item.

Output

Print the maximum obtainable value as a floating-point number.

Your answer will be accepted if it has absolute or relative error at most 10^{-3} .

Constraints

- $1 \leq n \leq 2 \cdot 10^5$
- $1 \leq C \leq 10^6$
- $1 \leq v_i, w_i \leq 10^6$
- If the exact answer is written as a reduced fraction p/q , then both p and q fit in signed 64-bit integers.

Example 1

Input

```
3 3
8 6
9 4
10 5
```

Output

```
6.75
```

Explanation

The second item has the highest value density: $9/4$. With capacity 3, take $3/4$ of that item and gain $(3/4) \cdot 9 = 27/4$.

Example 2

Input

```
4 8
6 3
15 5
14 7
9 6
```

Output

21.0

Explanation

Take the first two items completely. Their total weight is $3 + 5 = 8$, and their total value is $6 + 15 = 21$.