Lottery

Input file:	*standard input*
Output file:	*standard output*
Time limit:	0.5 seconds
Memory limit:	1024 MB

After working for over a year in a corporation, Peter realised that this is not the way to get rich. Instead, he quit his job and decided to participate in a lottery.

The lottery consists of a bag of N prizes. The value of the ith prize is $value[i]$ and its weight is $weight[i]$. A prize is chosen randomly from the bag according to its weight. That is, the probability of the ith prize to be chosen is $\frac{weight[i]}{\sum_{k=1}^{N} weight[k]}$. Now, Peter has a choice to make. He can accept the prize and go home with it, or he can refuse the prize. If Peter refuses, the prize is placed back in the bag and Peter plays another round. However, Peter can refuse at most K times. If he refuses K times, then he is forced to take home the prize from the $K + 1^{\text{th}}$ round.

Help Peter get rich and compute for him the maximum expected value he can get if he plays optimally.

Input

The first line of the input contains the integers N $(1 \leq N \leq 10^5)$ and K $(0 \leq K \leq 10^7)$, denoting the number of prizes in the bag and the maximum number of times Peter can refuse the prize.

Each of the following N lines contains 2 integers $value[i]$ and $weight[i]$ $(1 \leq value[i], weight[i] \leq 10^5)$, denoting the value and the weight of the ith prize.

Output

On a single line, print the maximum expected value Peter can get if he plays optimally. Your answer will be considered correct if the absolute or relative difference between your answer and judge's answer is less than 10^{-8}.

Example

standard input	standard output
3 2 1 4 2 3 3 1	2.093750000000