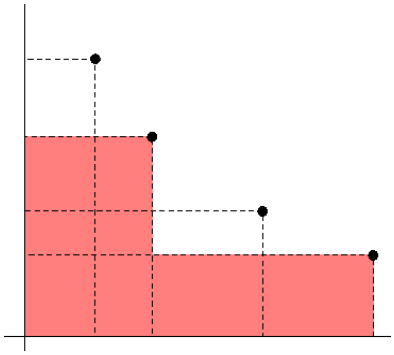
Problem B - Too many points

Description

Consider a set of n distinct points in the plane, $p_1, ..., p_n$, with a particular structure: no point exists in this set having larger or equal coordinate values than any other point in the set for both x and y-axis. The area *covered* by a point corresponds to the area of the rectangle bounded from above by the point and bounded from below by the origin.

For a given $k \leq n$, the goal is to choose a subset of k points that maximizes the union of the area covered by those points. See the following figure for an example with $n=4$ and $k=2$; the pink region is the union of the area covered by two of the four points.



The goal is to develop a bottom-up dynamic programming algorithm to solve this problem. Explore the following notion of subproblem:

P(i,j) is the subproblem of finding a subset of i points from the set $p_j, ..., p_n$ that contains p_j and that maximizes the union of the area covered by those points.

Input

The first line of each test case gives the value of n ($n \leq 1200$) and k, as positive integers. Then, in the following n lines, each point is described by its x and y coordinates as non-negative reals with 12 digits of precision.

Output

For each test case, print the maximum area as described above with 12 digits of precision. Note: Only bottom-up dynamic programming approaches will be considered correct for this problem. This will only be evaluated after the deadline. We suggest you to write the main working principles of your approach as a comment in your code.

Example

Example input:

```
5 3
0.376508963445 0.437693410334
```

```
0.948798695015 0.352125307881
0.176318878234 0.493630156084
0.029394902328 0.951299438575
0.235041868262 0.438197791997
```

Example output:

```
0.381410589193
```