ACM ICPC

UM Practice Contest 2

September 23

Problems

A	Maximal Rectangle
B	Best Time for Stock
C	Longest Increasing Subsequence
D	Optimal Binary Search Tree

Problem A

Maximal Rectangle

Description

Given a 2D binary matrix filled with 0's and 1's, find the largest rectangle containing only 1's and return its area.

For example, given the following matrix:

```
1 0 1 0 0

1 0 1 1 1

1 1 1 1 1

1 0 0 1 0
```

Return 6.

Input

The first line contains two integers n, m ($1 \leq n, m \leq 2000$).

The following n lines, each contains a binary string of length m.

Output

A single integer, represents the maximum size of the 1 rectangle.

Sample Input

```
4 5
```

```
10100
10111
11111
10010
```

Sample Output

```
6
```

Problem B

Best Time for Stock

Description

Say you have an array for which the ith element is the price of a given stock on day i. Design an algorithm to find the maximum profit. You may complete at most k transactions.

You may not engage in multiple transactions at the same time (ie, you must sell the stock before you buy again).

Input

The first line contains two integers n, k ($1 \leq n \leq 2000$).

The following n lines contains n integers, represents the stock price for n days.

All integers are positive and smaller than 10000.

For B-easy, $k = 1$.

For B-hard, $1 \leq k \leq 2000$.

Output

Output the maximum profit.

Sample Input

5 1

1 2 3 4 5

Sample Output

4

Problem C

Longest Increasing Subsequence

Description

The longest Increasing Subsequence (LIS) problem is to find the length of the longest subsequence of a given sequence such that all elements of the subsequence are sorted in increasing order. For example, length of LIS for { 10, 22, 9, 33, 21, 50, 41, 60, 80 } is 6 and LIS is {10, 22, 33, 50, 60, 80}.

Input

First line contains an integer n, which represents the length of the sequence.

The following line contains n integers in the sequence: $s[1], s[2], \dots, s[n]$.

For all inputs, $0 \leq s[i] \leq 100000000$

For C-easy, $1 \leq n \leq 2000$.

For C-hard, $1 \leq n \leq 200000$.

Output

The length of the longest increasing subsequence.

Sample Input

9

10 22 9 33 21 50 41 60 80

Sample Output

6

Problem D

Optimal Binary Search Tree

Description

An optimal binary search tree (BST) is a binary search tree which provides the smallest possible search time for a given access probabilities.

Given N nodes with access probabilities $(p_1, \ldots, p_n)$ in order, find a binary search tree such that the in-order traversal of the BST is the given order, and for all possible resulting depth $(d_1, \ldots, d_n), \sum_{i=1}^{n} d_i \cdot p_i$ is the minimum. The root of the tree has depth 1.

Input

The first line contains an integer n.

The next line contains n real number $p_1, \ldots, p_n (0 \leq p_i \leq 1)$. Note that the sum of all probabilities may not always equal to 1, but it does not matter.

For D-easy, $1 \leq n \leq 200$.

For D-hard, $1 \leq n \leq 2000$.

Output

A single real number represents the minimum sum, with exactly two digits to the right of the decimal point.

Sample Input

3
0 0.05 0.950

Sample Output

1.05

Note

The followings are the only five possible layout of the tree. The minimum possible search time is $0.95 \times 1 + 0.05 \times 2 = 1.05$.