

# SEGTREE - Segment Tree

It was Arbor Day. Alice implemented an RB-tree ([http://en.wikipedia.org/wiki/Red-black\\_tree](http://en.wikipedia.org/wiki/Red-black_tree)), Bob composed a segment tree ([http://en.wikipedia.org/wiki/Segment\\_tree](http://en.wikipedia.org/wiki/Segment_tree)), I made a binary tree ([http://en.wikipedia.org/wiki/Binary\\_tree](http://en.wikipedia.org/wiki/Binary_tree)) - we all have a bright outlook.

Lambda is always making mistakes while implementing segment trees (See his history of submissions). He then decides to draw a "segment tree". He puts  $n$  points on a plane, link certain pairs of them to form segments and all the segments form a tree. As a normal tree, it satisfies the following conditions:

1. Consider points as vertices, segments as edges, it forms a rooted tree ([http://en.wikipedia.org/wiki/Tree\\_%28graph\\_theory%29](http://en.wikipedia.org/wiki/Tree_%28graph_theory%29)).
2. Each node  $u$  is **strictly higher** than its parent, namely  $y_u > y_{parent\_of\_u}$ .
3. Segments may only intersect on their endpoints.

Lambda wants to minimize the total length of segments. **The tree can be rotated to satisfy above conditions.**

## Input

First line of input contains single integer  $n$  ( $1 \leq n \leq 500$ ).

Next  $n$  lines each contain two integers  $x_i, y_i$  denoting the coordinate of  $i$ -th point ( $0 \leq x_i, y_i \leq 1000$ ). Points are distinct.

## Output

The one and only line contains a real number representing the minimum length.

Your answer must be rounded up to 4 digits after the decimal point.

## Example 1

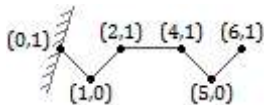
Input:

```
6
0 1
1 0
2 1
4 1
5 0
6 1
```

Output:

```
7.6569
```

Illustration:



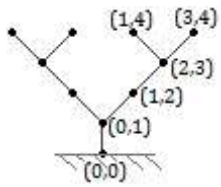
## Example 2

**Input:**

```
10
0 0
0 1
1 2
2 3
1 4
3 4
-1 2
-2 3
-1 4
-3 4
```

**Output:**

```
12.3137
```

**Illustration:**

This is just a sample test case. There's no negative in the real test data.