DA VYNCY

CHALLENGE DESCRIPTION:

You were reading The Da Vyncy Code, the translation of a famous murder mystery novel into Python. The Code is finally revealed on the last page. You had reached the second to last page of the novel, and then you went to take a bathroom break.

While you were in the bathroom, the Illuminati snuck into your room and shredded the last page of your book. You had 9 backup copies of the book just in case of an attack like this, but the Illuminati shredded the last page from each of the those books, too. Then they propped up a fan, aimed it at the remains, and turned it on at high-speed.

The last page of your book is now in tatters.

However, there are many text fragments floating in the air. You enlist an undergraduate student for a 'summer research project' of typing up these fragments into a file. Your mission: reassemble the last page of your book.

Problem Description
=============

(adapted from a problem by Julie Zelenski)

Write a program that, given a set of fragments (ASCII strings), uses the following method (or a method producing identical output) to reassemble the document from which they came:

At each step, your program searches the collection of fragments. It should find the pair of fragments with the maximal overlap match and merge those two fragments. This operation should decrease the total number of fragments by one. If there is more than one pair of fragments with a maximal overlap, you may break the tie in an arbitrary fashion.Fragments must overlap at their start or end. For example:

```
- "ABCDEF" and "DEFG" overlap with overlap length 3
- "ABCDEF" and "XYZABC" overlap with overlap length 3
- "ABCDEF" and "BCDE" overlap with overlap length 4
- "ABCDEF" and "XCDEZ" do *not* overlap (they have matching characters in the middle, but the overlap does not extend
to the end of either string).
```

Fear not - any test inputs given to you will satisfy the property that the tie-breaking order will not change the result, as long as you only ever merge maximally-overlapping fragments. Bonus points if you can come up with an input for which this property does not hold (ie, there exists more than 1 different final reconstruction, depending on the order in which different maximal-overlap merges are performed) -- if you find such a case, submit it in the comments to your code!

All characters must match exactly in a sequence (case-sensitive). Assume that your undergraduate has provided you with clean data (i.e., there are no typos).

INPUT SAMPLE:

Your program should accept as its first argument a path to a filename. Each line in this file represents a test case. Each line contains fragments separated by a semicolon, which your assistant has painstakingly transcribed from the shreds left by the Illuminati. You may assume that every fragment has length at least 2 and at most 1022 (excluding the trailing newline, which should *not* be considered part of the fragment). E.g. Here are two test cases.

```
O draconia;conian devil! Oh la;h lame sa;saint!
```

```
m quaerat voluptatem.;pora incidunt ut labore et d;, consectetur, adipisci velit;olore magnam aliqua;idunt ut labore
et dolore magn;uptatem.;i dolorem ipsum qu;iquam quaerat vol;psum quia dolor sit amet, consectetur, a;ia dolor sit
amet, conse;squam est, qui do;Neque porro quisquam est, qu;aerat voluptatem.;m eius modi tem;Neque porro qui;, sed
quia non numquam ei;lorem ipsum quia dolor sit amet;ctetur, adipisci velit, sed quia non numq;unt ut labore et dolore
magnam aliquam qu;dipisci velit, sed quia non numqua;us modi tempora incid;Neque porro quisquam est, qui dolorem i;uam
eius modi tem;pora inc;am al
```

OUTPUT SAMPLE:

Print out the original document, reassembled. E.g.

```
O draconian devil! Oh lame saint!
```

```
Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam
eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem.
```