After he had some fun, Ivancho decided to do something more productive. Now he started looking at binary numbers. He took a piece of paper and wrote two binary numbers *a* and *b* (may contain leading zeros). He wonders if he can convert *a* into *b* by performing operations of two types:

* Change a bit from number *a* with its opposite (i.e., replace 0 with 1 or 1 with 0);
* Swap any pair of bits from *a*.

After he thought for a while, he figured out that this is possible, but now he wonders what is the minimum number of operations that are needed in order to convert *a* into *b*. Help him with the task, so he can finally rest after the long day.

**Input**

The first and the second line of the file conversion.in contains *a* and *b* (may contain leading zeros), *a* and *b* have equal lengths.

**Output**

In the output file conversion.out print one number – the minimum number of operations needed to convert *a* into *b*.

**Constraints**

The lengths of *a* and *b are bigger than 1 and d*o not exceed 105

**Time limit: 1.0 sec**

**Memory limit: 256 MB**

**Example test**

|  |  |
| --- | --- |
| **Input (conversion.in)** | **Output (conversion.out)** |
| 01  10 | 1 |
| 110  100 | 1 |