

Encryption

An English text needs to be encrypted using the following encryption scheme. First, the spaces are removed from the text. Let L be the length of this text. Then, characters are written into a grid, whose rows and columns have the following constraints:

- $\lfloor \sqrt{L} \rfloor \leq \text{rows} \leq \text{column} \leq \lceil \sqrt{L} \rceil$, where $\lfloor x \rfloor$ is floor function and $\lceil x \rceil$ is ceil function

For example, the sentence `if man was meant to stay on the ground god would have given us roots` after removing spaces is `ifmanwasmeanttosstayonthe groundgodwouldhavegivenusroots` after removing spaces is `ifmanwasmeanttosstayonthe groundgodwouldhavegivenusroots` characters long, so it is written in the form of a grid with 7 rows and 8 columns.

```
ifmanwas  
meanttos  
stayonthe  
groundgo  
dwouldha  
vegivenu  
sroots
```

- Ensure that $\text{rows} \times \text{columns} \geq L$
- If multiple grids satisfy the above conditions, choose the one with the minimum area, i.e. $\text{rows} \times \text{columns}$.

The encoded message is obtained by displaying the characters in a column, inserting a space, and then displaying the next column and inserting a space, and so on. For example, the encoded message for the above rectangle is:

`imtgdvs fearwer mayoogo anouuio ntnnlvt wttdes aohghn sseoau`

You will be given a message in English with no spaces between the words. The maximum message length can be `81` characters. Print the encoded message.

Here are some more examples:

Sample Input:

```
haveaniceday
```

Sample Output:

```
hae and via ecy
```

Sample Input:

```
feedthedog
```

Sample Output:

```
fto ehg ee dd
```

Sample Input:

chillout

Sample Output:

clu hlt io