

# Zig-Zag Cipher

Input file:            **standard input**  
Output file:           **standard output**  
Time limit:            1 second  
Memory limit:          256 megabytes

The latest development in cryptography is the zig-zag cipher.

To encrypt a message using the zig-zag cipher, the letters of the message are written on the page in a zig-zag pattern. This is illustrated below for the message **CARTOGRAPHY**.

```
  C      O      P
  A  T  G  A  H
    R      R      Y
```

The letters are then read one row at a time, from the top row down to the bottom. In the diagram above, the first row spells **COP**, the second spells **ATGAH** and the third spells **RRY**. Finally these are combined to obtain the final encrypted message, which is **COPATGAHRRY**.

The zig-zag cipher can be extended to use as many rows as you like. For instance, using a zig-zag cipher with six rows, the message **UNSWPROGCHALSCONTESTONE** is encrypted to become **UAONHLTNSCSSEWGCEPOOTRN** as seen in the following diagram.

```
  U              A              O
    N      H  L      T  N
      S      C      S      S      E
        W      G      C      E
          P  O      O  T
            R              N
```

Your task is to write a program that can decrypt messages written in the zig-zag cipher. Your program must read the number of rows used in the cipher as well as an encrypted message, and must output the original (decrypted) message.

## Input

The first line of input will consist of a single integer  $r$ , the number of rows used in the zig-zag cipher ( $2 \leq r \leq 100$ ).

Following this will be the encrypted message. The encrypted message will consist entirely of upper-case letters, with no spaces or punctuation. The message will have at least 1 letter and at most 10 000 letters.

## Output

Your output should consist of a single line containing the entire decrypted message.

## Examples

standard input	standard output
3 COPATGAHRRY	CARTOGRAPHY
6 UAONHLTNSCSSEWGCEPOOTRN	UNSWPROGCHALSCONTESTONE