

Consider the following ciphering algorithm:

- For each character replace it with its code.
- Concatenate all of the obtained numbers.

Given a ciphered string, return the initial one if it is known that it consists only of lowercase letters.

**Note:** here the *character's code* means its decimal ASCII code, the numerical representation of a character used by most modern programming languages.

Example

For cipher = "10197115121" , the output should be  
decipher(cipher) = "easy" .

Explanation: charCode('e') = 101 , charCode('a') = 97 , charCode('s') = 115 and charCode('y') = 121 .

Input/Output

- [execution time limit] 4 seconds (js)
- [input] string cipher

A non-empty string which is guaranteed to be a cipher for some other string of lowercase letters.

*Guaranteed constraints:*  
 $2 \leq \text{cipher.length} \leq 100$  .

- [output] string

[JavaScript (ES6)] Syntax Tips

```
// Prints help message to the console
// Returns a string
function helloWorld(name) {
  console.log("This prints to the console when you Run Tests");
  return "Hello, " + name;
}
```