Caesar Cipher

Julius Caesar protected his confidential information by encrypting it using a cipher. Caesar's cipher shifts each letter by a number of letters. If the shift takes you past the end of the alphabet, just rotate back to the front of the alphabet. In the case of a rotation by 3, w, x, y and z would map to z, a, b and c.

```
Original alphabet:      abcdefghijklmnopqrstuvwxyz
Alphabet rotated +3:    defghijklmnopqrstuvwxyzabc
```

Example
$s = \texttt{There's-a-starman-waiting-in-the-sky}$
$k = 3$

The alphabet is rotated by 3, matching the mapping above. The encrypted string is $\texttt{Wkhuh'v-d-vwdupdq-zdlwlqj-lq-wkh-vnb}$.

Note: The cipher *only* encrypts letters; symbols, such as `-`, remain unencrypted.

Function Description

Complete the *caesarCipher* function in the editor below.

caesarCipher has the following parameter(s):

- *string s*: cleartext
- *int k*: the alphabet rotation factor

Returns

- *string:* the encrypted string

Input Format

The first line contains the integer, n, the length of the unencrypted string.
The second line contains the unencrypted string, s.
The third line contains k, the number of letters to rotate the alphabet by.

Constraints

$1 \leq n \leq 100$
$0 \leq k \leq 100$
s is a valid ASCII string without any spaces.

Sample Input

```
11
middle-Outz
2
```

Sample Output

```
okffng-Qwvb
```

Explanation

```
Original alphabet:      abcdefghijklmnopqrstuvwxyz
Alphabet rotated +2:    cdefghijklmnopqrstuvwxyzab

m -> o
i -> k
d -> f
d -> f
l -> n
e -> g
-    -
O -> Q
u -> w
t -> v
z -> b
```

Contest ends in a day

Submissions: 283
Max Score: 50
Difficulty: Easy

Rate This Challenge:

☆☆☆☆☆

More

Current Buffer (saved locally, editable)

C++14

```cpp
#include <bits/stdc++.h>

using namespace std;

// Complete the caesarCipher function below.
string caesarCipher(string s, int k) {


}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

    int n;
    cin >> n;
    cin.ignore(numeric_limits<streamsize>::max(), '\n');

    string s;
    getline(cin, s);

    int k;
    cin >> k;
    cin.ignore(numeric_limits<streamsize>::max(), '\n');

    string result = caesarCipher(s, k);

    fout << result << "\n";

    fout.close();

    return 0;
}

```

Line: 1 Col: 1

Upload Code as File   □ Test against custom input   Run Code   Submit Code