Q   Search   jigarpandya_ce ⌄

Array rotation - III   🔒 locked

Problem	Submissions	Leaderboard	Discussions

rotate_array function should rotate given array by number of elements mentioned. Please note that number of rotations for even and odd character could be different.

Complete code for *rotate_array* function and call it from main function to achieve following output from given input. Please do not change number of arguments or their types. Also do not change return type.

Input Format

Input will contain 2 lines.

First line will contain 1 string of characters (may have space and tabs).

Second line will have a number n and number m. Characters at even index needs to be rotated n times. While characters at odd index needs to be rotated by m times.

Constraints

0 < length of string <= 10000

0 < number of characters to rotate <= 1000000

Output Format

Output should have only one line containing string of characters (including space and tabs from input).

For example for following input

DDU STUDENTS

2 3

Characters at even indices are as follow:

D-U-S-U-E-T

When you rotate characters at even indices 2 times. You would get

E-T-D-U-S-U

Characters at odd indices are

-D- -T-D-N-S

When you rotate characters at odd indices 3 times. You would get

-D-N-S-D- -T

When you combine two rotated strings you will get output as follow

EDTNDSUDS UT

Sample Input 0

```
DDU STUDENTS
2 3
```

Sample Output 0

```
EDTNDSUDS UT
```

Sample Input 1

```
EDTNDSUDS UT
4 3
```

Sample Output 1

```
DDU STUDENTS
```

Sample Input 2

```
DDUNSSUDE TT
0 9
```

Sample Output 2

```
DDU STUDENTS
```

Submissions: 102
Max Score: 10
Difficulty: Hard

Rate This Challenge:
☆ ☆ ☆ ☆ ☆

More

Admin Options

✏ Edit Challenge

View Submissions

C ⌄

```c
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <stdlib.h>

void rotate_array(char str[], int even_index_rotations, int odd_index_rotations)
{
    // Write code here to rotate characters at even index by even_index_rotations
    // and characters at odd index by odd_index_rotations in the array.
    // Consider index starting from 0, i.e. first character in the array is at 0 index.
    // Use array traversing method.
    // Do not use array indexation method or pointer offset method.
    // Look at example for more clarity
```

```
14  }
15
16▼ int main() {
17
18▼     /* Enter your code here. Read input from STDIN. Print output to STDOUT */
19      return 0;
20  }
```

Line: 1 Col: 1

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

Interview Prep | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy |