Codility_

CodeCheck Report: trainingY3UYZP-QB6

Test Name:

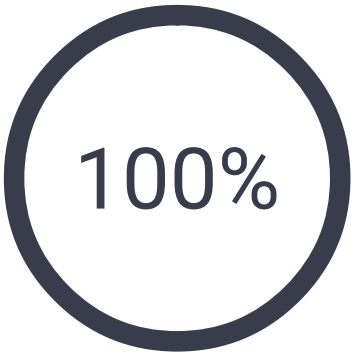
Check out Codility training tasks

Summary Timeline AI Assistant Transcript

Tasks summary

Task	Time spent	Score
CyclicRotation C# ⚠	2 min	100%

Total score



Tasks Details

1. **CyclicRotation**

Easy

Rotate an array to the right by a given number of steps.

Task Score	Correctness	Performance
100%	100%	Not assessed

Task description

An array A consisting of N integers is given. Rotation of the array means that each element is shifted right by one index, and the last element of the array is moved to the first place. For example, the rotation of array A = [3, 8, 9, 7, 6] is [6, 3, 8, 9, 7] (elements are shifted right by one index and 6 is moved to the first place).

The goal is to rotate array A K times; that is, each element of A will be shifted to the right K times.

Write a function:

```
class Solution { public int[] solution(int[] A,
int K); }
```

that, given an array A consisting of N integers and an integer K, returns the array A rotated K times.

For example, given

```
A = [3, 8, 9, 7, 6]
K = 3
```

the function should return [9, 7, 6, 3, 8]. Three rotations were made:

Solution

Programming language used:	C#
Total time used:	2 minutes
Effective time used:	2 minutes
Notes:	*not defined yet*

Task timeline

10:52:21 10:53:45

Code: 10:53:44 UTC, cs, final, score: **100** show code in pop-up

```
1  using System;
2  // you can also use other imports, for example:
```

```
[3, 8, 9, 7, 6] -> [6, 3, 8, 9, 7]
[6, 3, 8, 9, 7] -> [7, 6, 3, 8, 9]
[7, 6, 3, 8, 9] -> [9, 7, 6, 3, 8]
```

For another example, given

```
A = [0, 0, 0]
K = 1
```

the function should return [0, 0, 0]

Given

```
A = [1, 2, 3, 4]
K = 4
```

the function should return [1, 2, 3, 4]

Assume that:

- N and K are integers within the range [0..100];
- each element of array A is an integer within the range [−1,000..1,000].

In your solution, focus on **correctness**. The performance of your solution will not be the focus of the assessment.


```
3  // using System.Collections.Generic;
4
5  // you can write to stdout for debugging purposes,
6  // Console.WriteLine("this is a debug message");
7
8  class Solution {
9      static void ReversePart(int[] array, int start
10     {
11         while (start < end)
12         {
13             int temp = array[start];
14             array[start] = array[end];
15             array[end] = temp;
16             start++;
17             end--;
18         }
19     }
20     public int[] solution(int[] A, int K) {
21         // Implement your solution here
22         if(A.Length==0) return A;
23         K=K%A.Length;
24         if(K==0)
25         return A;
26         int temp=A.Length-K;
27         ReversePart(A,0,temp-1);
28         ReversePart(A,temp,A.Length-1);
29         ReversePart(A,0,A.Length-1);
30         return A;
31     }
32 }
```

Analysis summary

The solution obtained perfect score.

Analysis

expand all	Example tests	
▶	example first example test	✓ OK
▶	example2 second example test	✓ OK
▶	example3 third example test	✓ OK
expand all	Correctness tests	
▶	extreme_empty empty array	✓ OK
▶	single one element, 0 <= K <= 5	✓ OK
▶	double two elements, K <= N	✓ OK
▶	small1 small functional tests, K < N	✓ OK
▶	small2 small functional tests, K >= N	✓ OK
▶	small_random_all_rotations small random sequence, all rotations, N = 15	✓ OK
▶	medium_random medium random sequence, N = 100	✓ OK
▶	maximal maximal N and K	✓ OK