https://app.codility.com/demo/results/trainingWFMUPP-F75/

https://app.codility.com/demo/results/training5M39JZ-QHB/

https://app.codility.com/demo/results/trainingTYV2XF-PYR/

https://app.codility.com/demo/results/trainingA7YZQN-YAS/

https://app.codility.com/demo/results/training48HBDX-EB8/

https://app.codility.com/demo/results/trainingE7NFFC-6W3/

https://app.codility.com/demo/results/trainingJFJPEN-GWH/

https://app.codility.com/cert/view/cert9UX3WW-SDCUZ53U7N2FN9BH/details/

https://app.codility.com/demo/results/trainingMQVEPN-RRG/

https://app.codility.com/demo/results/training45HZ2H-F3A/

https://app.codility.com/demo/results/trainingRNBC9M-25T/

## **Tasks Details**

1.
BinaryGap

Find longest sequence of zeros in binary representation of an integer.

quence of Task Score Correctness Performance os in 100% Not assessed

A binary gap within a positive integer N is any maximal sequence of consecutive zeros that is surrounded by ones at both ends in the binary representation of N.

For example, number 9 has binary representation 1001 and contains a binary gap of length 2. The number 529 has binary representation 1000010001 and contains two binary gaps: one of length 4 and one of length 3. The number 20 has binary representation 10100 and contains one binary gap of length 1. The number 15 has binary representation 1111 and has no binary gaps. The number 32 has binary representation 100000 and has no binary gaps.

Write a function:

int solution(int N);

that, given a positive integer N, returns the length of its longest binary gap. The function should return 0 if N doesn't contain a binary gap.

For example, given N = 1041 the function should return 5, because N has binary representation 10000010001 and so its longest binary gap is of length 5. Given N = 32 the function should return 0, because N has binary representation '100000' and thus no binary gaps.

Write an **efficient** algorithm for the following assumptions:

 N is an integer within the range [1..2,147,483,647].

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

## Solution

Programming language used: C++

Total time used: 7 minutes

Fffective time used: 7 minutes

Notes: not defined yet

## Task timeline

## **Tasks Details**

steps.

CyclicRotation

Rotate an array to the right by a given number of Correctness

100%

Performance

100% Not assessed

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).

Task Score

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:

vector<int> solution(vector<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:

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:

## Solution

Programming language used: C++20

Total time used: 12 minutes

Effective time used: 12 minutes

Notes: not defined yet

## Task timeline

- 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.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

## **Tasks Details**

Check out Codility training tasks

Easy

# 1. OddOccurrencesInArray

Task Score

Correctness

Performance

Find value that occurs in odd number of elements.

100%

100%

100%

A non-empty array A consisting of N integers is given. The array contains an odd number of elements, and each element of the array can be paired with another element that has the same value, except for one element that is left unpaired.

For example, in array A such that:

$$A[0] = 9 \quad A[1] = 3 \quad A[2] = 9$$

$$A[3] = 3 \quad A[4] = 9 \quad A[5] = 7$$

$$A[6] = 9$$

- the elements at indexes 0 and 2 have value 9,
- the elements at indexes 1 and 3 have value 3.
- the elements at indexes 4 and 6 have value 9.
- the element at index 5 has value 7 and is unpaired.

Write a function:

that, given an array A consisting of N integers fulfilling the above conditions, returns the value of the unpaired element.

For example, given array A such that:

$$A[0] = 9$$
  $A[1] = 3$   $A[2] = 9$   
 $A[3] = 3$   $A[4] = 9$   $A[5] = 7$ 

A[6] = 9

the function should return 7, as explained in the example above.

Write an efficient algorithm for the following assumptions:

- N is an odd integer within the range [1..1,000,000];
- each element of array A is an integer within the range [1..1,000,000,000];
- all but one of the values in A occur an even number of times.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

## Solution

Programming language used: C++20

Total time used: 6 minutes

Effective time used: 6 minutes

Notes: not defined yet

## Task timeline

## Tasks Details

1.

## **FrogJmp**

Count

minimal number of jumps from

position X to Y.

Task Score

Correctness

Performance

100%

100%

100%

A small frog wants to get to the other side of the road. The frog is currently located at position X and wants to get to a position greater than or equal to Y. The small frog always jumps a fixed distance, D.

Count the minimal number of jumps that the small frog must perform to reach its target.

Write a function:

int solution(int X, int Y, int D);

that, given three integers X, Y and D, returns the minimal number of jumps from position X to a position equal to or greater than Y.

For example, given:

X = 10

Y = 85

D = 30

the function should return 3, because the frog will be positioned as follows:

- after the first jump, at position 10 + 30
  = 40
- after the second jump, at position 10 + 30 + 30 = 70
- after the third jump, at position 10 + 30
   + 30 + 30 = 100

Write an **efficient** algorithm for the following assumptions:

 X, Y and D are integers within the range [1..1,000,000,000];

## Solution

Programming language used: C++

Total time used: 6 minutes

Effective time used: 6 minutes

Notes: not defined yet

## Task timeline

X ≤ Y.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

## Tasks Details

1. **P∈** ĝ Fir

## PermMissingElem

Task Score

Correctness

Performance

Find the missing element in a given

permutation.

100%

100%

100%

An array A consisting of N different integers is given. The array contains integers in the range [1..(N + 1)], which means that exactly one element is missing.

Your goal is to find that missing element.

Write a function:

int solution(vector<int> &A);

that, given an array A, returns the value of the missing element.

For example, given array A such that:

A[0] = 2

A[1] = 3

A[2] = 1

A[3] = 5

the function should return 4, as it is the missing element.

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [0..100,000];
- the elements of A are all distinct;
- each element of array A is an integer within the range [1..(N + 1)].

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

## Solution

Programming language used: C++20

STTZU

Total time used:

5 minutes

Effective time used:

5 minutes

Notes:

not defined yet

Task timeline

## **Tasks Details**

TapeEquilibrium

Minimize the value Task Score Correctness Performance |(A[0] + ... + A[P-1])| |(A[0] + ... + A[N-1])| |(A[0] + ... + A[N-1])|

A non-empty array A consisting of N integers is given. Array A represents numbers on a tape.

Any integer P, such that 0 < P < N, splits this tape into two non-empty parts: A[0], A[1], ..., A[P - 1] and A[P], A[P + 1], ..., A[N - 1].

The difference between the two parts is the value of: |(A[0] + A[1] + ... + A[P-1]) - (A[P] + A[P+1] + ... + A[N-1])|

In other words, it is the absolute difference between the sum of the first part and the sum of the second part.

For example, consider array A such that:

A[0] = 3

A[1] = 1

A[2] = 2

A[3] = 4

A[4] = 3

We can split this tape in four places:

• P = 1, difference = |3 - 10| = 7

• P = 2, difference = |4 - 9| = 5

• P = 3, difference = |6 - 7| = 1

• P = 4, difference = |10 - 3| = 7

Write a function:

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

that, given a non-empty array A of N integers, returns the minimal difference that can be achieved.

For example, given:

A[0] = 3

A[1] = 1

A[2] = 2

A[3] = 4

A[4] = 3

the function should return 1, as explained above.

## Solution

Programming language used: C#

Total time used: 5 minutes

Effective time used: 5 minutes

Notes: not defined yet

## Task timeline

Write an **efficient** algorithm for the following assumptions:

- N is an integer within the range [2..100,000];
- each element of array A is an integer within the range [-1,000..1,000].

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

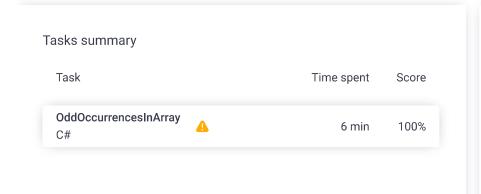
## Codility\_

## CodeCheck Report: trainingRNBC9M-25T

Test Name:

Check out Codility training tasks

Al Assistant Transcript Summary Timeline





## Tasks Details

1. **OddOccurrencesInArray** Find value that occurs in odd number of elements.

Task Score

Correctness

100%

Performance

100%

## Task description

A non-empty array A consisting of N integers is given. The array contains an odd number of elements, and each element of the array can be paired with another element that has the same value, except for one element that is left unpaired.

For example, in array A such that:

$$A[0] = 9$$
  $A[1] = 3$   $A[2] = 9$ 

$$A[3] = 3 \quad A[4] = 9 \quad A[5] = 7$$

A[6] = 9

- the elements at indexes 0 and 2 have value 9,
- the elements at indexes 1 and 3 have value 3,
- the elements at indexes 4 and 6 have value 9,
- the element at index 5 has value 7 and is unpaired.

Write a function:

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

## Solution

Programming language used: C#

Total time used: 6 minutes

100%

Effective time used: 6 minutes

Notes: not defined yet

Task timeline



that, given an array A consisting of N integers fulfilling the above conditions, returns the value of the unpaired element.

For example, given array A such that:

```
A[0] = 9 A[1] = 3 A[2] = 9

A[3] = 3 A[4] = 9 A[5] = 7

A[6] = 9
```

the function should return 7, as explained in the example above.

Write an efficient algorithm for the following assumptions:

- N is an odd integer within the range [1..1,000,000];
- each element of array A is an integer within the range [1..1,000,000,000];
- all but one of the values in A occur an even number of times.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

#### Test results - Codility

Code: 11:51:10 UTC, cs, final,

```
score: 100
 1
     using System;
     // you can also use other imports, for example:
 2
3
     using System.Collections.Generic;
 5
     // you can write to stdout for debugging purposes,
     // Console.WriteLine("this is a debug message");
 6
8
     class Solution {
         public int solution(int[] A) {
10
             // Implement your solution here
11
             var hash = new HashSet<int>();
             foreach(var element in A)
12
13
                 if(hash.Contains(element))
                     hash.Remove(element);
14
15
                     hash.Add(element);
16
             foreach (var i in hash)
17
18
                 return i;
19
             return 0;
20
         }
21
     }
```

show code in pop-up

## Analysis summary

The solution obtained perfect score.

## Analysis

expar	nd all	Example tes	ts	
<b>&gt;</b>	example1 example test		✓ OK	
expar	nd all	Correctness te	ests	
<b>&gt;</b>	simple1 simple test n=5		✓ OK	
<b>&gt;</b>	simple2 simple test n=11		✓ OK	
•	extreme_single_ [42]	item	✓ OK	
<b>&gt;</b>	small1 small random test r	n=201	✓ OK	
<b>&gt;</b>	small2 small random test r	n=601	✓ OK	
expar	nd all	Performance to	ests	
•	medium1 medium random tes	st n=2,001	✓ OK	
<b>&gt;</b>	medium2	st n=100,003	✓ OK	

## Test results - Codility

	big1	✓ OK
	big random test n=999,999, multiple	
	repetitions	
<b></b>	big2	✓ OK
	big random test n=999,999	

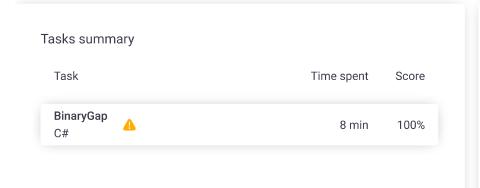
## Codility\_

## CodeCheck Report: trainingMQVEPN-RRG

Test Name:

Check out Codility training tasks

Summary Timeline 💩 AI Assistant Transcript





## **Tasks Details**

## 1. BinaryGap

Easy

Find longest sequence of zeros in binary representation of an integer.

Task Score

100%

Correctness

Performance

100% Not assessed

## Task description

A binary gap within a positive integer N is any maximal sequence of consecutive zeros that is surrounded by ones at both ends in the binary representation of N.

For example, number 9 has binary representation 1001 and contains a binary gap of length 2. The number 529 has binary representation 1000010001 and contains two binary gaps: one of length 4 and one of length 3. The number 20 has binary representation 10100 and contains one binary gap of length 1. The number 15 has binary representation 1111 and has no binary gaps. The number 32 has binary representation 100000 and has no binary gaps.

Write a function:

class Solution { public int solution(int N); }

that, given a positive integer N, returns the length of its longest binary gap. The function should return 0 if N doesn't contain a

## Solution

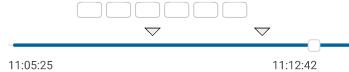
Programming language used: C#

Total time used: 8 minutes

Effective time used: 8 minutes

Notes: not defined yet

Task timeline



binary gap.

For example, given N = 1041 the function should return 5, because N has binary representation 10000010001 and so its longest binary gap is of length 5. Given N = 32 the function should return 0, because N has binary representation '100000' and thus no binary gaps.

Write an efficient algorithm for the following assumptions:

• N is an integer within the range [1..2,147,483,647].

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

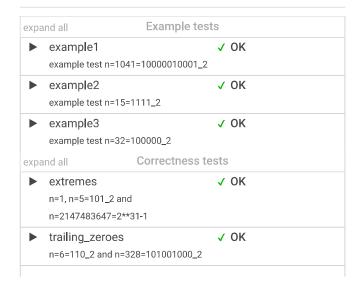
#### Test results - Codility

```
Code: 11:12:42 UTC, cs, final,
                                      show code in pop-up
 score: 100
1
     using System;
 2
     // you can also use other imports, for example:
     // using System.Collections.Generic;
 4
5
     // you can write to stdout for debugging purposes,
 6
     // Console.WriteLine("this is a debug message");
8
     class Solution {
9
         public int solution(int N) {
10
              // Implement your solution here
              bool isFound = false;
11
12
              int len = 0, max = 0;
13
14
              for(uint i = 1; i<=N; i <<= 1)</pre>
15
              {
16
                  if((i & N) != 0)
17
                  {
                      if(isFound)
18
19
                      {
                          if(len > max)
20
21
                          {
22
                              max = len;
23
24
                          len = 0;
25
                      isFound = true;
26
27
                  }
28
                  else if(isFound)
29
                  {
30
                      ++len;
31
                  }
32
              }
33
              return max;
34
35
36
```

## Analysis summary

The solution obtained perfect score.

#### **Analysis**



## Test results - Codility

iest i	esults - Codility	
•	power_of_2 n=5=101_2, n=16=2**4 and n=1024=2**10	√ 0K
•	simple1 n=9=1001_2 and n=11=1011_2	√ OK
•	simple2 n=19=10011 and n=42=101010_2	✓ OK
•	simple3 n=1162=10010001010_2 and n=5=101_2	√ OK
•	medium1 n=51712=110010100000000_2 and n=20=10100_2	√ ok
•	medium2 n=561892=10001001001011100100_2 and n=9=1001_2	<b>√</b> 0K
•	medium3 n=66561=1000001000000001_2	√ OK
•	large1 n=6291457=1100000000000000000000000000000000000	√ OK
•	large2 n=74901729=10001110110111010001 1100001	√ OK
•	large3 n=805306373=110000000000000000000000000000000000	√ OK
•	large4 n=1376796946=101001000010000010 0000100010010_2	√ OK
•	large5 n=1073741825=1000000000000000000000000000000000000	√ OK
•	large6 n=1610612737=110000000000000000000000000000000000	√ OK

## Codility\_

## CodeCheck Report: training45HZ2H-F3A

Test Name:

Check out Codility training tasks

Summary Timeline 💩 Al Assistant Transcript

asks summary		
Task	Time spent	Score
CyclicRotation C#	31 min	100%



## **Tasks Details**

1. CyclicRotation<br/>Rotate an array to the rightTask ScoreCorrectnessPerformanceby a given number of steps.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:

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

For example, given

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

## Solution

Programming language used: C#

Total time used: 31 minutes

Effective time used: 31 minutes

Notes: not defined yet

Task timeline

11:14:10

11:44:59

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

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.

Copyright 2009–2024 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

#### Test results - Codility

```
Code: 11:44:58 UTC, cs, final,
                                    show code in pop-up
 score: 100
 1
     using System;
 2
     // you can also use other imports, for example:
3
     // using System.Collections.Generic;
 4
 5
     // you can write to stdout for debugging purposes,
     // Console.WriteLine("this is a debug message");
 6
8
     class Solution {
         public int[] solution(int[] A, int K) {
 9
10
             // Implement your solution here
             var rotatedArr = new int[A.Length];
11
12
13
             if(rotatedArr.Length > 0)
14
             {
                  for(int i=-1, j=A.Length - (K%A.Length)
15
16
17
                     rotatedArr[i] = A[(j+i)%A.Length];
18
19
             }
20
21
             return rotatedArr;
22
         }
23
     }
```

## Analysis summary

The solution obtained perfect score.

## Analysis

expa	nd all	Example tests	S	
<b>&gt;</b>	example		<b>√</b> 0I	K
	first example test			
<b>•</b>	example2		<b>√</b> 0I	K
	second example test	t		
<b>•</b>	example3		<b>√</b> 0I	K
	third example test			
expa	nd all	Correctness tes	sts	
•	extreme_empty		<b>√</b> 0I	K
	empty array			
<b>•</b>	single		<b>√</b> 0I	K
	one element, 0 <= K	<= 5		
<b>&gt;</b>	double		<b>√</b> 0I	K
	two elements, K <= N	1		
<b>•</b>	small1		<b>√</b> 0I	K
	small functional test	s, K < N		
•	small2		<b>√</b> 0I	K
	small functional test	s, K >= N		
•	small_random_a	II_rotations	<b>√</b> 0l	K
	small random seque	nce, all rotations, N		
	= 15			
•	medium_random	1	<b>√</b> 0I	K
	medium random seq	uence, N = 100		

maximal N and K

✓ OK