

Tasks Details

[Check out Codility training tasks](#)

Medium	1. MinAvgTwoSlice	Task Score	Correctness	Performance
	Find the minimal average of any slice containing at least two elements.			
		100%	100%	100%

Task description

A non-empty array A consisting of N integers is given. A pair of integers (P, Q) , such that $0 \leq P < Q < N$, is called a *slice* of array A (notice that the slice contains at least two elements). The *average* of a slice (P, Q) is the sum of $A[P] + A[P + 1] + \dots + A[Q]$ divided by the length of the slice. To be precise, the average equals $(A[P] + A[P + 1] + \dots + A[Q]) / (Q - P + 1)$.

For example, array A such that:

```
A[0] = 4
A[1] = 2
A[2] = 2
A[3] = 5
A[4] = 1
A[5] = 5
A[6] = 8
```

contains the following example slices:

- slice (1, 2), whose average is $(2 + 2) / 2 = 2$;
- slice (3, 4), whose average is $(5 + 1) / 2 = 3$;
- slice (1, 4), whose average is $(2 + 2 + 5 + 1) / 4 = 2.5$.

The goal is to find the starting position of a slice whose average is minimal.

Write a function:

```
def solution(A)
```

that, given a non-empty array A consisting of N integers, returns the starting position of the slice with the minimal average. If there is more than one slice with a minimal average, you should return the smallest starting position of such a slice.

For example, given array A such that:

```
A[0] = 4
A[1] = 2
A[2] = 2
A[3] = 5
A[4] = 1
A[5] = 5
A[6] = 8
```

the function should return 1, as explained above.

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

Solution

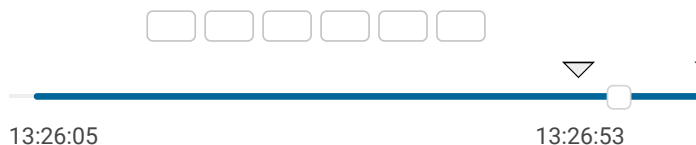
Programming language used: Python

Total time used: 1 minutes ?

Effective time used: 1 minutes ?

Notes: *not defined yet*

Task timeline



Code: 13:26:53 UTC, py, final,
score: 100

[show code in pop-up](#)

```
1 # you can write to stdout for debugging purposes, e.g.
2 # print("this is a debug message")
3
4 def solution(A):
5     min_idx = 0
6     min_value = 10001
7
8     for idx in range(0, len(A)-1):
9         if (A[idx] + A[idx+1])/2.0 < min_value:
10             min_idx = idx
11             min_value = (A[idx] + A[idx+1])/2.0
12         if idx < len(A)-2 and (A[idx] + A[idx+1] + A[idx+2])/3.0 < min_value:
13             min_idx = idx
14             min_value = (A[idx] + A[idx+1] + A[idx+2])/3
15
16     return min_idx
```

Analysis summary

The solution obtained perfect score.

- N is an integer within the range [2..100,000];
- each element of array A is an integer within the range [−10,000..10,000].

Copyright 2009–2021 by Codility Limited. All Rights Reserved. Unauthorized copying, publication or disclosure prohibited.

Analysis

Detected time complexity: **O(N)**

expand all	Example tests	
▶	example example test	✓ OK
expand all	Correctness tests	
▶	double_quadruple two or four elements	✓ OK
▶	simple1 simple test, the best slice has length 3	✓ OK
▶	simple2 simple test, the best slice has length 3	✓ OK
▶	small_random random, length = 100	✓ OK
▶	medium_range increasing, decreasing (length = ~100) and small functional	✓ OK
expand all	Performance tests	
▶	medium_random random, N = ~700	✓ OK
▶	large_ones numbers from -1 to 1, N = ~100,000	✓ OK
▶	large_random random, N = ~100,000	✓ OK
▶	extreme_values all maximal values, N = ~100,000	✓ OK
▶	large_sequence many sequences, N = ~100,000	✓ OK

The PDF version of this report that may be downloaded on top of this site may contain sensitive data including personal information. For security purposes, we recommend you remove it from your system once reviewed.