

Яндекс. Тренировки по алгоритмам 2.0, занятие 1 (B)

D. Строительство школы

Ограничение времени	1 секунда
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

В деревне Интернетовка все дома расположены вдоль одной улицы по одну сторону от нее. По другую сторону от этой улицы пока ничего нет, но скоро все будет – школы, магазины, кинотеатры и т.д.

Для начала в этой деревне решили построить школу. Место для строительства школы решили выбрать так, чтобы суммарное расстояние, которое проезжают ученики от своих домов до школы, было минимально.

План деревни можно представить в виде прямой, в некоторых целочисленных точках которой находятся дома учеников. Школу также разрешается строить только в целочисленной точке этой прямой (в том числе разрешается строить школу в точке, где расположен один из домов – ведь школа будет расположена с другой стороны улицы).

Напишите программу, которая по известным координатам домов учеников поможет определить координаты места строительства школы.

Формат ввода

Сначала вводится число N — количество учеников ($0 < N < 100001$). Далее идут в строго возрастающем порядке координаты домов учеников — целые числа, не превосходящие 2×10^9 по модулю.

Формат вывода

Выведите одно целое число — координату точки, в которой лучше всего построить школу. Если ответов несколько, выведите любой из них.

Пример 1

Ввод <input type="text"/>	Вывод <input type="text"/>
4 1 2 3 4	3

Пример 2

Ввод <input type="text"/>	Вывод <input type="text"/>
3 -1 0 1	0

```
1 # считываем данные
2 N = int(input().strip()) # количество учеников
3 houses = list(map(int, input().split())) # координаты домов учеников
4 norm_houses = [house - houses[0] for house in houses]
5
6 cur_houses_left = 0
7 cur_houses_right = N - 1
8 min_dist = sum(norm_houses)
9 best_idx = 0
10 cur_dist = min_dist
11 for i in range(1, N):
12     diff = norm_houses[i] - norm_houses[i - 1]
13     cur_houses_left += 1
14     cur_dist = cur_dist + (cur_houses_left - cur_houses_right) * diff
15     cur_houses_right -= 1
16     if cur_dist < min_dist:
17         min_dist = cur_dist
18         best_idx = i
19
20 print(houses[best_idx])
```