

Тренировки по алгоритмам 4.0 от Яндекса — Разминка

🕒 11 апр 2024, 19:29:13
старт: 30 окт 2023, 12:00:00
финиш: 27 ноя 2023, 00:59:59
длительность: 27д. 12ч.
...

Объявления жюри

📘 Ваше участие в соревновании завершено. Вы можете дорешивать задачи и отправлять решения вне соревнования

Положение участников Задачи Посылки

Ф. Лифт

Ограничение времени	1 секунда
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Тридцатого декабря все сотрудники известной IT-компании отправляются праздновать Новый год! На парковке офиса сотрудников уже ожидают автобусы, чтобы отвезти их в ресторан. Известно, что на i -м этаже работает a_i сотрудников, а парковка расположена на нулевом этаже. Изначально лифт расположен на этаже с парковкой. Какое минимальное количество времени лифт будет перевозить всех людей на парковку? Известно, что лифт движется со скоростью один этаж в секунду, а посадка и высадка происходит мгновенно.

Формат ввода

В первой строке дано единственное целое число $k(1 \leq k \leq 10^9)$ — количество людей, которое вмещает лифт за одну поездку. Во второй строке дано единственное целое число n — количество этажей в здании. В следующих $n(1 \leq n \leq 10^5)$ строках дано по одному целому неотрицательному числу $a_i(0 \leq a_i \leq 10^9)$, которое обозначает количество сотрудников на этаже номер i .

Формат вывода

Выведите единственное целое число — минимальное количество секунд, которое необходимо, чтобы все сотрудники оказались на парковке.

Пример

Ввод	Вывод
2 3 3 0 1	8

Язык Python 3.9 (PyPy 7.3.11) ▾

Набрать здесь Отправить файл

```
1 def correct_Man_On_Floor(stack):
2     while stack and stack[-1] == 0:
3         stack.pop()
4
5 fin = open('input.txt')
6 K = int(fin.readline())
7 N = int(fin.readline())
8
9 stack = []
10 for _ in range(N):
11     stack.append(int(fin.readline()))
12
13 correct_Man_On_Floor(stack)
14
15 def answer(max_man_in_elevator, stack):
16     res_time = 0
17     while stack:
18         res_time += len(stack)*2
19
20         if stack[-1] > max_man_in_elevator:
21             count = (stack[-1] // max_man_in_elevator)-1
22             res_time += len(stack)*2*count
23             stack[-1] -= max_man_in_elevator*count
24
25         curr_man_in_elevator = 0
26         while (curr_man_in_elevator < max_man_in_elevator) and stack:
27             if stack[-1]+curr_man_in_elevator <= max_man_in_elevator:
28                 curr_man_in_elevator += stack.pop()
29                 correct_Man_On_Floor(stack)
30             else:
31                 stack[-1] -= (max_man_in_elevator-curr_man_in_elevator)
32                 curr_man_in_elevator = max_man_in_elevator
33
34     return res_time
35
36 print(answer(K, stack))
```

Отправить 📘 осталось 100 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
1 ноя 2023, 06:29:28	95470585	F	Python 3.9 (PyPy 7.3.11)	OK	-	270ms	28.32Mb	-	-	отчёт
1 ноя 2023, 06:28:33	95470577	F	Python 3.9 (PyPy 7.3.11)	OK	-	278ms	28.55Mb	-	-	отчёт
31 окт 2023, 21:18:04	95427583	F	Python 3.9 (PyPy 7.3.11)	OK	-	286ms	28.52Mb	-	-	отчёт
31 окт 2023, 21:16:00	95427265	F	Python 3.9 (PyPy 7.3.11)	WA	-	199ms	28.10Mb	6	-	отчёт