

Тренировки по алгоритмам 3.0 от Яндекса — Дивизион В

🕒 4 апр 2024, 02:38:36

📅 старт: 28 окт 2023, 21:14:17

🏁 финиш: 26 ноя 2023, 03:14:16

⌚ длительность: 28д. 5ч.

...

Объявления жюри

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

Положение участников

Задачи

Посылки

3. Коллекционер Диего

Ограничение времени	2 секунды
Ограничение памяти	256Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

Диего увлекается коллекционированием наклеек. На каждой из них написано число, и каждый коллекционер мечтает собрать наклейки со всеми встречающимися числами. Диего собрал N наклеек, некоторые из которых, возможно, совпадают. Как-то раз к нему пришли K коллекционеров. i -й из них собрал все наклейки с номерами не меньшими, чем p_i . Напишите программу, которая поможет каждому из коллекционеров определить, сколько недостающих ему наклеек есть у Диего. Разумеется, гостей Диего не интересуют повторные экземпляры наклеек.

Формат ввода

В первой строке содержится единственное число N ($0 \leq N \leq 100\,000$) — количество наклеек у Диего. В следующей строке содержатся N целых неотрицательных чисел (не обязательно различных) — номера наклеек Диего. Все номера наклеек не превосходят 10^9 . В следующей строке содержится число K ($0 \leq K \leq 100\,000$) — количество коллекционеров, пришедших к Диего. В следующей строке содержатся K целых чисел p_i ($0 \leq p_i \leq 10^9$), где p_i — наименьший номер наклейки, не интересующий i -го коллекционера.

Формат вывода

Для каждого коллекционера в отдельной строке выведите количество **различных** чисел на наклейках, которые есть у Диего, но нет у этого коллекционера.

Пример 1

Ввод	Вывод
1	0
5	1
2	
4 6	

Пример 2

Ввод	Вывод
3	3
100 1 50	0
3	2
300 0 75	

Язык

Python 3.9 (PyPy 7.3.11)

▼

Набрать здесь

Отправить файл

```
1 fin = open('input.txt')
2 N = int(fin.readline())
3
4 stickers = set([int(x) for x in fin.readline().split()])
5 stickers = sorted(stickers)
6
7 def bsearch(array, left, right, val):
8     while left < right:
9         mid = (left+right)//2
10        # TTTT, FFFF, TTFF, search for first F
11        if array[mid] < val:
12            left = mid+1
13        else:
14            right = mid
15
16    if left == len(array)-1 and array[left] < val: left += 1
17    return left
18
19 K = int(fin.readline())
20 answers = [None]*K
21 for i, num in enumerate([int(x) for x in fin.readline().split()]):
22     answers[i] = str(bsearch(stickers, 0, len(stickers)-1, num))
23
24 print('\n'.join(answers))
```

Отправить

📘 осталось 100 попыток

Предыдущая

Следующая

Время посылки	ID	Задача	Компилятор	Вердикт	Тип посылки	Время	Память	Тест	Баллы	
15 ноя 2023, 18:06:29	97421947	3	Python 3.9 (PyPy 7.3.11)	OK	-	350ms	58.77Mb	-	-	отчёт
15 ноя 2023, 18:02:55	97421377	3	Python 3.9 (PyPy 7.3.11)	WA	-	190ms	28.09Mb	4	-	отчёт

- ✓

1. Гистограмма
- ✓

2. Красивая строка
- ✓

3. Коллекционер Диего
- ✓

4. Контрольная работа
- ✓

5. Хорошая строка
- ✓

6. Операционные системы lite
- ✓

7. SNTP
- ✓

8. Минимальный прямоугольник
- ✓

9. Сумма в прямоугольнике
- ✓

10. Скучная лекция
- ✓

11. Стек с защитой от ошибок
- ✓

12. Правильная скобочная последовательность
- ✓

13. Постфиксная запись
- ✓

14. Сортировка вагонов lite
- ✓

15. Великое Лайнландское переселение
- ✓

16. Очередь с защитой от ошибок
- ✓

17. Игра в пьяницу
- ✓

18. Дек с защитой от ошибок
- ✓

19. Хипуй
- ✓

20. Пирамидальная сортировка
- ✗

21. Три единицы подряд
22. Кузнечик
23. Калькулятор
24. Покупка билетов
25. Гвоздики
26. Самый дешёвый путь
27. Вывести маршрут максимальной стоимости
28. Ход конём
29. Кафе
30. НОП с восстановлением ответа
- ✓

31. Поиск в глубину
- ✓

32. Компоненты связности
- ✓

33. Списывание
- ✓

34. Топологическая сортировка
- ✓

35. Поиск цикла
- ✓

36. Длина кратчайшего пути
- ✓

37. Путь в графе
- ✓

38. Блохи
- ✓

39. Путь спелеолога
- ✓

40. Метро