



Национальный исследовательский университет ИТМО
(Университет ИТМО)

Факультет систем управления и робототехники

Дисциплина: Алгоритмы и структуры данных
Отчет по практической работе (1604 задача).

Студент:
Евстигнеев Дмитрий
Группа: *R3242*
Преподаватель:
Тропченко Андрей Александрович

Санкт-Петербург
2021

Цель: написать программу для решения задачи №1604 на [сайте Timus Online](https://timusonline.com/)

Задача:

1604. В Стране Дураков

Ограничение времени: 1.0 секунды

Ограничение памяти: 64 МБ

Главный бульдог-полицейский Страны Дураков решил ввести ограничение скоростного режима на автомобильной трассе, ведущей от Поля Чудес к пруду Черепахи Тортиллы. Для этого он заказал у Папы Карло n знаков ограничения скорости. Папа Карло слабо разбирался в дорожном движении и поэтому изготовил знаки с разными ограничениями на скорость: 49 км/ч, 34 км/ч, 42 км/ч, и т.д.



Всего получилось k различных ограничений: n_1 знаков с одним ограничением, n_2 знаков со вторым ограничением, и т.д. ($n_1 + \dots + n_k = n$)

Бульдог-полицейский ничуть не расстроился, получив такие знаки, напротив, он решил извлечь из этого экономическую выгоду. Дело в том, что по Правилам дорожного движения Страны Дураков ограничение на скорость действует вплоть до следующего знака. Если на знаке написано число 60, это означает, что участок от данного знака до следующего нужно проехать ровно со скоростью 60 километров в час — не больше и не меньше. Бульдог распорядился расставить знаки так, чтобы обогатившимся на Поле Чудес автолюбителям во время своего движения по трассе приходилось как можно больше раз менять скорость. Для этого нужно расставить имеющиеся знаки в правильном порядке. Если Вы поможете бульдогу это сделать, то он готов будет поделиться с Вами частью своих доходов.

Исходные данные

В первой строке дано число k — количество различных типов знаков с ограничением скорости ($1 \leq k \leq 10000$). Во второй строке через пробел перечислены целые положительные числа n_1, \dots, n_k . Сумма всех n_i не превосходит 10000.

Результат

Выведите n целых чисел в пределах от 1 до k — порядок, в котором нужно расставить по трассе имеющиеся знаки. Вне зависимости от того, какой знак стоит первым, считается, что, проезжая его, водитель меняет скорость, так как до этого ограничения не действовали. Если задача имеет несколько решений, выведите любое.

Пример

исходные данные	результат
2 2 2	1 2 1 2

Принято системой (JUDGE_ID: 231802FR):

ID	Дата	Автор	Задача	Язык
9290883	19:36:36 30 мар 2021	Dmitry Evstigneev	1604. В Стране Дураков	G++ 9.2 x64

Решение на языке C++:

```
#include <iostream>

using namespace std;

typedef struct {
    long count;
    long index;
} sign;

long k, min_c=LONG_MAX, max_c=LONG_MIN, kol_sign, first_sign, second_sign;
sign n[10000];

int main()
{
    cin >> k;
    for (long i=0; i<k; i++)
    {
        cin >> n[i].count;
        n[i].index = i+1;
        kol_sign += n[i].count;
        if (n[i].count >= max_c)
            { max_c = n[i].count; second_sign = i; }
        if (n[i].count < min_c && i != second_sign)
            { min_c = n[i].count; first_sign = i; }
    }

    while (kol_sign > 0)
    {
        if (n[second_sign].count > 0)
        {
            cout << n[second_sign].index << " ";
            n[second_sign].count--;
            kol_sign--;
        }
        if (n[first_sign].count > 0)
        {
            cout << n[first_sign].index << " ";
            n[first_sign].count--;
            kol_sign--;
        }

        min_c = LONG_MAX; max_c = LONG_MIN;
        for(long i=0; i < k; i++) if (n[i].count > 0)
        {
            if (n[i].count >= max_c)
                { max_c = n[i].count; second_sign=i; }
            if (n[i].count < min_c)
                { min_c = n[i].count; first_sign=i; }
        }
    }

    return 0;
}
```

Суть алгоритма:

На каждой итерации выбираем ячейку с максимальным и минимальным количеством знаков и выводим их последовательно. Формально не сортируя массив (не меняя положение элементов) мы выводим отсортированные по количеству элементов значения.

Примеры работы программы:

```
2
2 2
2 1 2 1
Process returned 0 (0x0)   execution time : 5.353 s
Press any key to continue.
```

```
3
4 3 2
1 3 2 3 1 2 1 2 1
Process returned 0 (0x0)   execution time : 88.535 s
Press any key to continue.
```