

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа № 13

по дисциплине «Алгоритмы и структуры данных»

Stepik – техника «Разделяй и властвуй» (разделы 6-7)

Подготовил:

студент группы Р3217

Бураков Илья Алексеевич

Преподаватели:

Романов Алексей Андреевич

Волчек Дмитрий Геннадьевич

Задача на программирование: двоичный поиск

В первой строке даны целое число $1 \leq n \leq 10^5$ и массив $A[1 \dots n]$ из n различных натуральных чисел, не превышающих 10^9 , в порядке возрастания, во второй — целое число $1 \leq k \leq 10^5$ и k натуральных чисел b_1, \dots, b_k , не превышающих 10^9 . Для каждого i от 1 до k необходимо вывести индекс $1 \leq j \leq n$, для которого $A[j] = b_i$, или -1 , если такого j нет.

Sample Input:

```
5 1 5 8 12 13
5 8 1 23 1 11
```

Sample Output:

```
3 1 -1 1 -1
```

Решение

```
arr = list(map(int, input().split()))
n = arr.pop(0)
_, *qs = map(int, input().split())

ans = []
for qv in qs:
    l = 0
    r = len(arr) - 1
    pivot = (l + r) // 2
    while arr[pivot] != qv and l != r + 1:
        if qv > arr[pivot]:
            l = pivot + 1
        else:
            r = pivot - 1
        pivot = (l + r) // 2

    ans.append(-1 if arr[pivot] != qv else pivot + 1)

print(" ".join(map(str, ans)))
```

Задача на программирование: число инверсий

Первая строка содержит число $1 \leq n \leq 10^5$, вторая — массив $A[1 \dots n]$, содержащий натуральные числа, не превосходящие 10^9 . Необходимо посчитать число пар индексов $1 \leq i < j \leq n$, для которых $A[i] > A[j]$. (Такая пара элементов называется инверсией массива. Количество инверсий в массиве является в некотором смысле его мерой неупорядоченности: например, в упорядоченном по неубыванию массиве инверсий нет вообще, а в массиве, упорядоченном по убыванию, инверсию образуют каждые два элемента.)

Sample Input:

```
5
2 3 9 2 9
```

Sample Output:

```
2
```

Решение

```
#include <iostream>
#include <vector>
#include <algorithm>
```

```

using namespace std;

long long count_inversions(vector<int> &arr, int left, int right) {
    // left subarray index is inclusive
    // right subarray index is exclusive (!)

    // length of the subarray to sort
    int subarray_n = right - left;

    if (subarray_n == 1) {
        // already sorted, job done
        return 0;
    }

    // beginning of left and right subarrays
    int left_beginning = left;
    int right_beginning = left + subarray_n / 2;

    // recursively sort subarrays
    long long result = 0;
    result += count_inversions(arr, left_beginning, right_beginning);
    result += count_inversions(arr, right_beginning, right);

    // merge sorted subarrays
    auto merged_arr = vector<int>(subarray_n);
    int i = left_beginning;
    int j = right_beginning;
    for (int k = 0; k < subarray_n; k++) {
        if (j == right || (i != right_beginning && arr[i] <= arr[j])) {
            // left subarray wins for current k
            merged_arr[k] = arr[i];
            i++;
        } else {
            // right subarray wins for current k
            merged_arr[k] = arr[j];
            j++;
            // Since right subarray won, all left subarray elements larger
            // than the element from
            // right subarray will form an inversion. Add their quantity to
            // result.
            result += right_beginning - i;
        }
    }

    // place merged_arr
    copy_n(merged_arr.begin(), subarray_n, arr.begin() + left);

    return result;
}

int main() {
    // read N
    int n;
    cin >> n;

    // read input array
    auto arr = vector<int>(n);
    for (auto &e : arr) {
        cin >> e;
    }

    // sorting
    cout << count_inversions(arr, 0, n);

    return 0;
}

```

Задача на программирование: сортировка подсчётом

Первая строка содержит число $1 \leq n \leq 10^4$, вторая — n натуральных чисел, не превышающих 10. Выведите упорядоченную по неубыванию последовательность этих чисел.

Sample Input:

```
5
2 3 9 2 9
```

Sample Output:

```
2 2 3 9 9
```

Решение

```
from collections import Counter

n = int(input())
print("".join(f" {c}" * num for c, num in sorted(Counter(map(int,
input().split())).items())).strip())
```