Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

Факультет цифровых трансформаций

Дисциплина:

«Алгоритмы и структуры данных»

Практическая работа №7

«Жадные алгоритмы»

Выполнила:

Абаянцева Е. Ю., студент группы Ј3110

Санкт-Петербург 2024 г.

- 1. Подсчёт памяти: два массива 16 байт, один int 4 байта. Вызов k раз (на запускаемом тесте 5 раз) с подсчётом памяти каждой из трёх функций (в сумме три массива в каждой функции 24 байта, два инта в двух 8 байтов). Всего в сумме 361 байт.
- 2. Подсчёт асимптотики сложность $O(k \cdot (m+n))$.
- 3. Код:

```
□class Solution {
            vector<int> maxnumber(vector<int>& nums1, vector<int>& nums2, int k) {
                int m = nums1.size(), n = nums2.size();
                 vector<int> nums;
                 for (int i = max(0, k - n); i <= min(k, m); i++) {
                    nums = max(nums, mergenums(maxnums(nums1, i), maxnums(nums2, k - i)));
13
14
15
16
                 return nums:
            vector<int> maxnums(vector<int>& nums, int k) {
                int i = (int)nums.size() - k;
19
                 vector<int> end_nums;
                for (int num : nums) {
    while (i > 0 && end_nums.size() > 0 && end_nums.back() < num) {
                         end_nums.pop_back();
24
25
                    end_nums.push_back(num);
27
                end_nums.resize(k);
                return end_nums;
31
32
            vector<int> mergenums(vector<int> nums1, vector<int> nums2) {
                 vector<int> nums;
                while ( nums1.size() > 0 || nums2.size() > 0 ) {
35
36
37
38
                         vector<int> &mas = nums1:
                         nums.push back(mas[0]):
                         mas.erase(mas.begin());
39
40
                     else {
                         vector<int> &mas = nums2;
                         nums.push_back(mas[0]);
43
                         mas.erase(mas.begin());
45
46
47
48
49
                return nums;
       };
      pint main() {
            vector<int> nums1 = { 3, 4, 6, 5 };
            vector<int> nums2 = { 9, 1, 2, 5, 8, 3 };
            int k = 5;
55
56
            Solution x:
            vector<int> result = x.maxnumber(nums1, nums2, k);
            for (int num : result) {
   cout << num << " ";</pre>
58
            return 0;
```

Рис. 1: КОД

- 4. Прохождение всех локальных тестов:
 - (a) Tect 1:

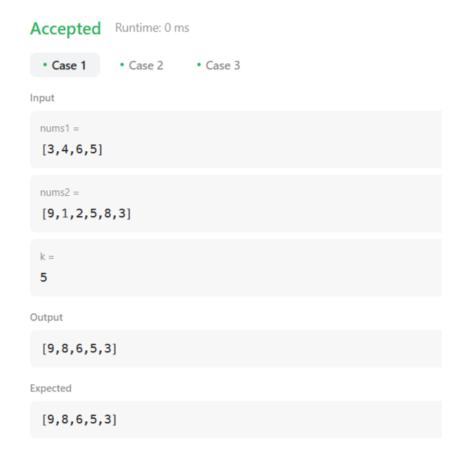


Рис. 2: тест 1

(b) Tect 2:

Accepted Runtime: 0 ms

Case 1

• Case 2

• Case 3

Input

nums1 =

[6,7]

nums2 =

[6,0,4]

k =

5

Output

[6,7,6,0,4]

Expected

[6,7,6,0,4]

Рис. 3: тест 2

(c) Tect 3:



Рис. 4: Тест 3

5. Прохождение основных тестов:



Рис. 5: ОСНОВНЫЕ ТЕСТЫ

6. В этой задаче требуется использовать жадный алгоритм, потому что задача требует на каждом шагу выбирать оптимально рабочий вариант (в нашем варианте составить список максимальных элементов).

На каждом шаге мы выбираем массив с максимальными числами, сравнивая его с составленным массивом с таким же количеством элементов, но с другими максимальными элементами (если они существуют).