

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

|  |  |
| --- | --- |
| Факультет | Информатика и вычислительная техника |
|  | (наименование факультета) |
| Кафедра | Кибербезопасность информационных систем |
|  | (наименование кафедры) |

**ОТЧЕТ по практической работе**

**по дисциплине “Методы программирования”**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Автор |  | | |  | Бобро М.Д. | | | |
|  | (подпись, дата) | | |  |  | | | |
| Обозначение | 10.05.01.550000.000 О | | Группа | | | | ВКБ31 |
| Направление подготовки | | 10.05.01 Компьютерная безопасность | | | | | |
| Профиль | Компьютерная безопасность | | | | | | |
| Преподаватель |  | | |  | | Савельев В.А. | |
|  | (подпись, дата) | | |  | |  | |

г. Ростов-на-Дону

2021 год

**Лабораторная работа №3**

Даны две последовательности, требуется найти и вывести их наибольшую общую подпоследовательность.

Входные данные

В первой строке входных данных содержится число NN – длина первой последовательности (1 ≤ NN ≤ 1000). Во второй строке заданы члены первой последовательности (через пробел) – целые числа, не превосходящие 10000 по модулю.

В третьей строке записано число MM – длина второй последовательности (1 ≤ MM ≤ 1000). В четвертой строке задаются члены второй последовательности (через пробел) – целые числа, не превосходящие 10000 по модулю.

Выходные данные

Требуется вывести наибольшую общую подпоследовательность данных последовательностей, через пробел.

**Примеры**

|  |  |
| --- | --- |
| входные данные  3  1 2 3  3  2 3 1 | выходные данные  2 3 |

#include <iostream>

#include <cmath>

#include <algorithm>

#include <vector>

#include <queue>

#include <stack>

#include <map>

#include <set>

#include <bitset>

using namespace std;

const int N = 1002;

int x[N], y[N], a[N][N];

int main() {

//ios\_base::sync\_with\_stdio(false);

//cin.tie(0);

//cout.tie(0);

int n, m;

cin >> n;

for (int i = 1; i <= n; i++) {

cin >> x[i];

}

cin >> m;

for (int i = 1; i <= m; i++) {

cin >> y[i];

}

for (int i = 1; i <= n; i++) {

for (int j = 1; j <= m; j++) {

if (x[i] == y[j]) {

a[i][j] = 1 + a[i - 1][j - 1];

}

else {

a[i][j] = max(a[i - 1][j], a[i][j - 1]);

}

}

}

vector<int> v;

int i = n, j = m;

while (i && j) {

if (x[i - 1] == y[j - 1]) {

v.push\_back(x[i]);

i--;

j--;

}

else if (a[i - 1][j] == a[i][j]) {

i--;

}

else {

j--;

}

}

for (int k = v.size() - 1; k >= 0; k--) {

cout << v[k] << ' ';

}

return 0;

}

