

**Федеральное государственное автономное образовательное учреждение
высшего образования
"Национальный исследовательский университет "Высшая школа экономики"**

Московский институт электроники и математики им. А.Н. Тихонова НИУ ВШЭ
Департамент компьютерной инженерии

Курс: Алгоритмизация и программирование

Раздел	Макс. оценка	Итоговая оценка
Работа программы	1	
Тесты	1	
Правильность алгоритма	3	
Ответы на вопросы	2	
Дополнительное задание	3	
Итого	10	

Отчет по лабораторной работе № 8

Студент: Чапайкин Арсений Георгиевич

Группа: БИВ242

Вариант: № 171 (25, 37)

Руководитель: Елисеенко А.М.

Оценка:

Дата сдачи:

Содержание

Задание	2
Листинг программы	3
Распечатка тестов к программе и результатов	6

Задание

1. Даны два одномерных массива целых чисел A и B . Вычислить одномерный массив C , который содержит отрицательные элементы, присутствующие в массиве B 1 раз, но отсутствующие в массиве A .
2. В заданной целочисленной прямоугольной матрице поменять местами два последних положительных элемента в нечетных столбцах.

Листинг программы

Задание 1

```
1 #include <iostream>
2 #include <map>
3 #include <utility>
4
5 std::pair<size_t, int*> task1(size_t lenA, size_t lenB, int* const a, int* const b)
6 {
7     size_t lenC = 0;
8     int* c = (int*)malloc(sizeof(int) * lenB);
9
10    std::map<int, int> cntB;
11    for (int* x = b; x != b + lenB; x++) {
12        cntB[*x]++;
13    }
14    std::map<int, int> cntA;
15    for (int* x = a; x != a + lenA; x++) {
16        cntA[*x]++;
17    }
18
19    for (int* x = b; x != b + lenB; x++) {
20        if (cntB[*x] == 1 && *x < 0 && cntA[*x] == 0) {
21            c[lenC++] = *x;
22        }
23    }
24
25    c = (int*)realloc(c, lenC);
26    return std::make_pair(lenC, c);
27 }
28
29 void printArray(size_t len, int* const array) {
30     for (int* x = array; x != array + len; x++) {
31         std::cout << *x << ' ';
32     }
33     std::cout << '\n';
34 }
35
36 int* inputArray(size_t len) {
37     int* array = (int*)new int[len];
38     for (int* x = array; x != array + len; x++) {
39         std::cin >> *x;
40     }
41     return array;
42 }
43
44 int main() {
45     size_t lenA;
46     std::cin >> lenA;
47
48     int* a = inputArray(lenA);
49
50     size_t lenB;
51     std::cin >> lenB;
52
53     int* b = inputArray(lenB);
54
55     std::pair<size_t, int*> res = task1(lenA, lenB, a, b);
```

```

55     if (res.second == nullptr) {
56         std::cout << "Resulting array is empty\n";
57     } else {
58         std::cout << "Resulting array:\n";
59         printArray(res.first, res.second);
60     }
61
62     free(res.second);
63     delete[] a;
64     delete[] b;
65
66     return 0;
67 }

```

Задание 2

```

1  #include <iostream>
2
3  void task2(size_t rows, size_t cols, int** const matrix) {
4      for (size_t j = 0; j < cols; j += 2) {
5          int* first = nullptr;
6          int* second = nullptr;
7
8          for (size_t i = 0; i < rows; i++) {
9              if (*(matrix + i) + j) > 0) {
10                 first = second;
11                 second = *(matrix + i) + j;
12             }
13         }
14
15         if (first != nullptr && second != nullptr) {
16             std::swap(*first, *second);
17         }
18     }
19 }
20
21 void printMatrix(size_t rows, size_t cols, int* const* const matrix) {
22     for (int* const* x = matrix; x != matrix + rows; x++) {
23         for (int* y = *x; y != *x + cols; y++) {
24             std::cout << *y << ' ';
25         }
26         std::cout << '\n';
27     }
28 }
29
30 int** inputMatrix(size_t rows, size_t cols) {
31     int** matrix = new int*[rows];
32     for (int** x = matrix; x != matrix + rows; x++) {
33         *x = new int[cols];
34         for (int* y = *x; y != *x + cols; y++) {
35             std::cin >> *y;
36         }
37     }
38     return matrix;
39 }
40
41 int main() {
42     size_t rows, cols;

```

```
43     std::cin >> rows >> cols;
44
45     int** matrix = inputMatrix(rows, cols);
46     task2(rows, cols, matrix);
47
48     std::cout << "Resulting matrix:\n";
49     printMatrix(rows, cols, matrix);
50
51     for (int** x = matrix; x != matrix + rows; x++) {
52         delete[] *x;
53     }
54     delete[] matrix;
55
56     return 0;
57 }
```

Распечатка тестов к программе и результатов

Задание 1

Номер	Исходные данные	Результат
1	4 1 2 3 4 4 -1 -2 -3 -4	Resulting array: -1 -2 -3 -4
2	4 1 2 3 4 4 -1 -1 -1 -1	Resulting array is empty
3	4 -1 -2 -3 -4 4 -1 -2 -3 -4	Resulting array is empty
4	4 1 2 3 4 4 1 2 3 4	Resulting array is empty
5	4 1 2 3 4 4 -1 0 -3 4	Resulting array: -1 -3

Задание 2

Номер	Исходные данные	Результат
1	3 3 1 2 3 4 5 6 7 8 9	Resulting matrix: 1 2 3 7 5 9 4 8 6
2	3 3 1 2 3 4 5 6 -7 -8 -9	Resulting matrix: 4 2 6 1 5 3 -7 -8 -9
3	3 3 1 2 3 -4 -5 -6 7 8 9	Resulting matrix: 7 2 9 -4 -5 -6 1 8 3
4	3 3 -1 -2 -3 -4 -5 -6 -7 -8 -9	Resulting matrix: -1 -2 -3 -4 -5 -6 -7 -8 -9
5	3 3 1 2 3 -4 -5 -6 0 0 0	Resulting matrix: 1 2 3 -4 -5 -6 0 0 0