Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«**Пермский национальный исследовательский политехнический университет»**

Кафедра «Информационные технологии и автоматизированные системы»

**ОТЧЕТ**

Дисциплина: «Основы алгоритмизации и программирования»

Тема : "Работа с массивами"

Вариант 9

Выполнила работу

Студентка группы РИС-22-1Б

Верхоланцева Екатерина Сергеевна

Проверила

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Г. Пермь-2022

**1. Постановка задачи.**

а) 1.Сформировать одномерный массив целых чисел, используя датчик слу чайных чисел.

2.Распечатать полученный массив.

3.Удалить первый элемент с заданным значением.

4.Сдвинуть массив циклически на К элементов вправо.

5.Распечатать полученный массив.

б) Написать функцию для вычисления суммы элементов квадратной матри- цы, которые расположены ниже главной диагонали. С ее помощью найти максимальное значение такой суммы в n матрицах.

**2. Анализ задачи.**

a) 1. Задать начальную точку отсчета случайных значений с помощью srand(time(NULL)).

2. С помощью цикла for заполнить массив из 100 элементов случайными числами.

3. Ввод пользователем длины массива, с которой нужно будет работать (не больше 100).

4. Вывод массива заданной длины циклом for.

5. Ввод пользователем значения, первое вхождение которого нужно будет удалить.

6. Циклом for поиск в массиве заданного значения, если нашлось, сдвинуть весь массив на 1 элемент влево, тем самым «затерев» нужный элемент. Вывести готовый массив до len - 1 элемента, чтобы массив стал на 1 элемент меньше.

7. Ввод пользователем числа k, на которое нужно циклически сдвинуть массив вправо.

8. Цикл for, каждая итерация которого будет сдвигать массив на 1 элемент вправо. Присвоить дополнительной переменной tmp1 нулевой элемент массива, на его место поставить значение последнего элемента массива. Внутренним циклом for сдвинуть весь массив на элемент вправо.

Внешний цикл for выполнится k раз.

9. Вывести полученный массив.

б) 1. Объявить функцию summ с параметрами n - размерность квадратной матрицы и arr[5][5] - самой матрицей. Функция будет пробегать по всем элементам матрицы, которые стоят ниже главной диагонали с помощью двух циклов for (внешний - по строкам (for (int i = 0; i < n; i++)), внутренний - по столбцам (for (int j = 0; j < i; j++))) и суммировать значения элементов.

2. В функции main ввести N - количество матриц, которые будут введены.

3. Задать двумерный массив arr[5][5] из 5 строк и 5 столбцов, n = 5.

4. С помощью трех циклов for (1-ый для количества матриц, 2-ой для строк, 3-ий для столбцов) ввести значения элементов матриц с клавиатуры построчно.

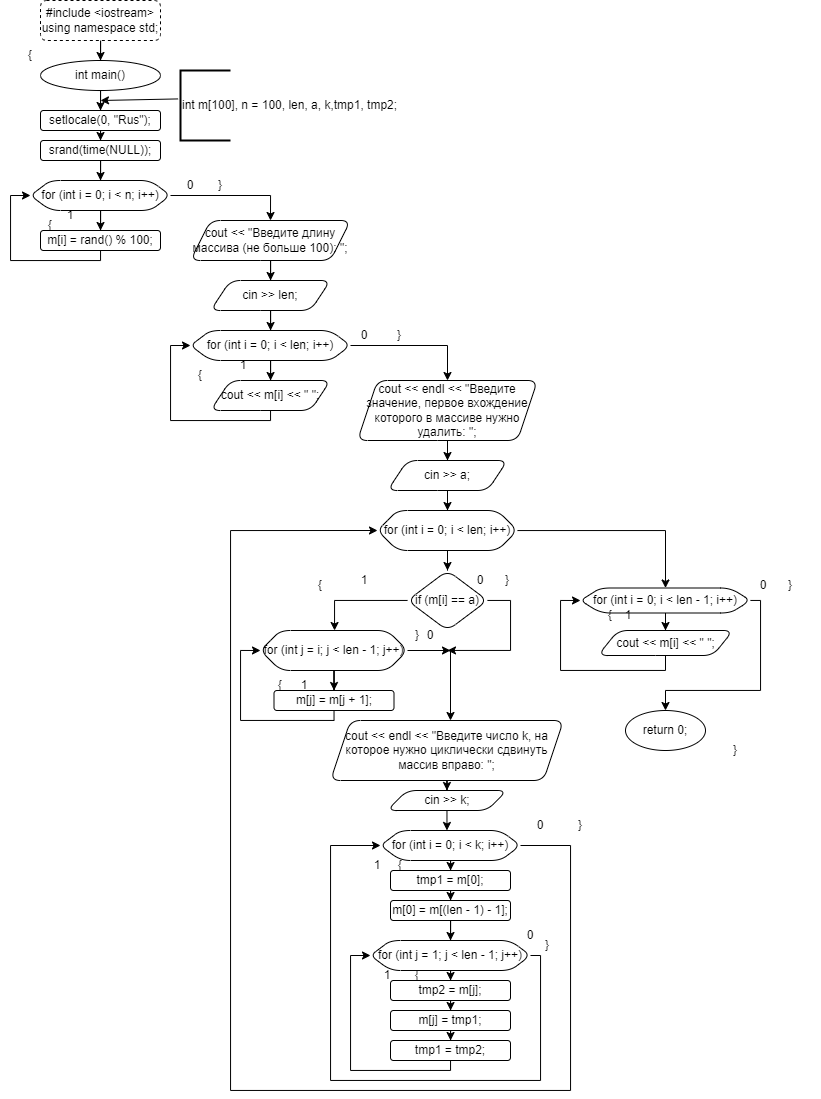
После каждой введенной матрицы выводить полученную матрицу для проверки.

Присваивать S результат функции summ для каждой матрицы, сравнивать это значение с максимальным, если больше, то максимальному присваи вать S, а также запоминать номер матрицы с максимальной суммой.

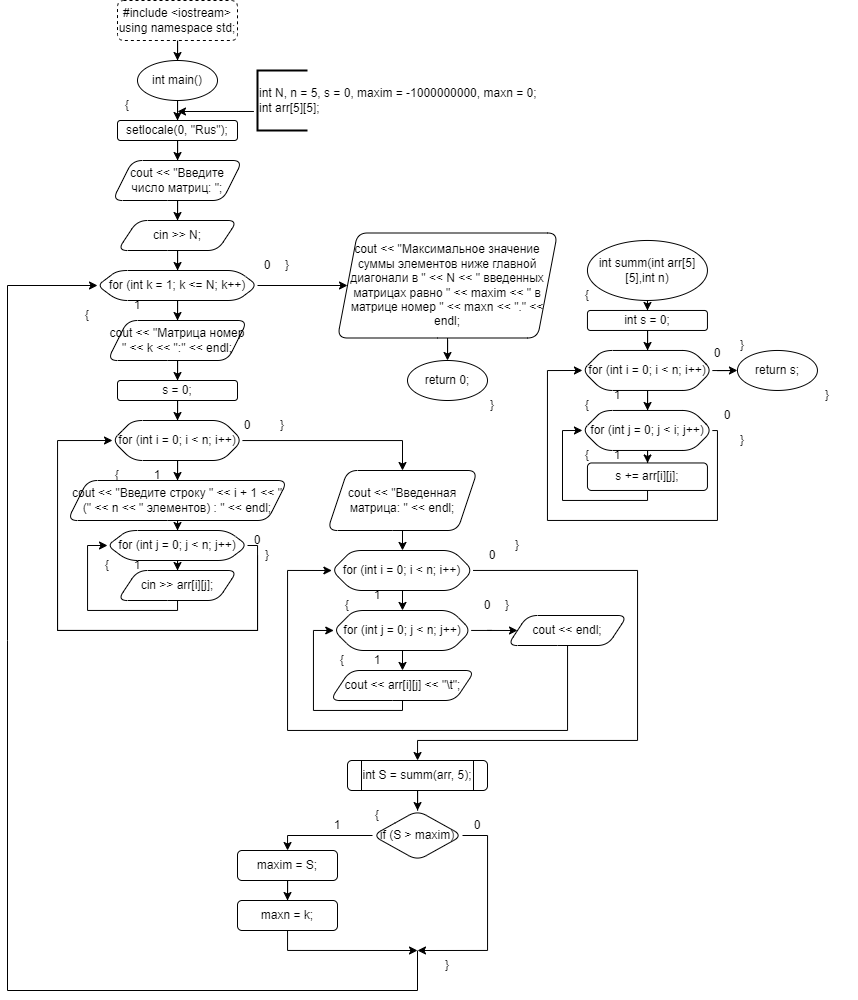
5. Выйдя из всех циклов, вывести максимальное значение суммы и номер матрицы с максимальной суммой.

**3. Блок-схема.**

а)



б)



**4. Код.**

**a)** #include <iostream>

using namespace std;

/\*

1.Сформировать одномерный массив целых чисел, используя датчик случайных чисел.

2.Распечатать полученный массив.

3.Удалить первый элемент с заданным значением.

4.Сдвинуть массив циклически на К элементов вправо.

5.Распечатать полученный массив.

\*/

int main() {

setlocale(0, "Rus");

srand(time(NULL));

int m[100], n = 100, len, a, k;

for (int i = 0; i < n; i++) { // заполнение массива случайными числами

m[i] = rand() % 100;

}

cout << "Введите длину массива (не больше 100): ";

cin >> len;

cout << endl;

cout << "{ " << " ";

for (int i = 0; i < len; i++){ // вывод массива

cout << m[i] << " ";

}

cout << " }" << " ";

cout << endl;

cout << endl << "Введите значение, первое вхождение которого в массиве нужно удалить: ";

cin >> a;

for (int i = 0; i < len; i++) {

if (m[i] == a) { // если нашли этот элемент - сдвигаем весь массив на 1 элемент влево,

for (int j = i; j < len - 1; j++) { // тем замым "затирая" заданный элемент

m[j] = m[j + 1];

}

}

}

cout << endl;

cout << "{ " << " ";

for (int i = 0; i < len - 1; i++) { // вывод массива на 1 элемент меньше, т.к. массив должен был стать на элемент меньше

cout << m[i] << " ";

}

cout << " }" << " ";

cout << endl;

int tmp1, tmp2;

cout << endl << "Введите число k, на которое нужно циклически сдвинуть массив вправо: ";

cin >> k;

for (int i = 0; i < k; i++) { // каждая итерация будет сдвигать на один элемент вправо

tmp1 = m[0]; // присваиваем tmp1 значение нулевого элемента

m[0] = m[(len - 1) - 1]; // на его место ставим значение последнего элемента

for (int j = 1; j < len - 1; j++) { // сдвигаем весь массив на элемент вправо

tmp2 = m[j];

m[j] = tmp1;

tmp1 = tmp2;

}

}

cout << endl;

cout << "{ " << " ";

for (int i = 0; i < len - 1; i++) { // вывод массива

cout << m[i] << " ";

}

cout << " }" << " ";

cout << endl;

return 0;

}

б) #include <iostream>

using namespace std;

/\*

Написать функцию для вычисления суммы элементов квадратной матрицы,

которые расположены ниже главной диагонали. С ее помощью найти

максимальное значение такой суммы в n матрицах.

\*/

int summ(int arr[5][5],int n)

{

int s = 0;

for (int i = 0; i < n; i++) {

for (int j = 0; j < i; j++) {

s += arr[i][j];

}

}

return s;

}

int main()

{

setlocale(0, "Rus");

int N, n = 5, s = 0, maxim = -1000000000, maxn = 0;

int arr[5][5];

cout << "Введите число матриц: ";

cin >> N;

for (int k = 1; k <= N; k++) { // ввод N матриц

cout << endl;

cout << "Матрица номер " << k << ":" << endl;

s = 0;

for (int i = 0; i < n; i++) {

cout << "Введите строку " << i + 1 << " (" << n << " элементов) : " << endl;

for (int j = 0; j < n; j++) {

cin >> arr[i][j];

}

}

cout << "Введенная матрица: " << endl;

for (int i = 0; i < n; i++) {

for (int j = 0; j < n; j++) {

cout << arr[i][j] << "\t";

}

cout << endl;

}

int S = summ(arr, 5);

if (S > maxim) {

maxim = S;

maxn = k;

}

}

cout << endl;

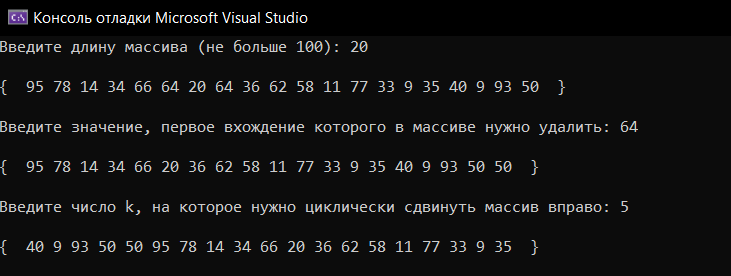
cout << "Максимальное значение суммы элементов ниже главной диагонали в " << N << " введенных матрицах равно " << maxim << " в матрице номер " << maxn << "." << endl;

return 0;

}

**5. Скриншот вывода.**

**а)**



**б)**

