Отчёт по лабораторной работе №8 «Пользовательские функции»

# Вариант №1

## Титульный лист

Лабораторная работа №8 «Пользовательские функции»  
Студент: \_\_\_\_\_\_\_\_\_\_\_\_  
Группа: \_\_\_\_\_\_\_\_\_\_\_\_  
Дата выполнения: \_\_\_\_\_\_\_\_\_\_\_\_

## Цель работы

Познакомиться с механизмом составления и организации взаимодействия пользовательских функций языка С.

## Текст задания

Переписать программу «Двумерные массивы» (вариант 1), разделив её на несколько функций:  
- В функции main() организовать интерфейс взаимодействия с пользователем посредством текстового меню;  
- Все операции ввода-вывода и вычислительные операции оформить как отдельные функции с параметрами и/или возвращаемым значением.

## Назначение идентификаторов (переменных и функций)

- inputMatrix() – функция ввода матрицы;  
- countNonZeroRows() – функция подсчёта строк без нулевых элементов;  
- findMaxRepeatingElement() – функция поиска максимального повторяющегося элемента;  
- main() – главная функция с интерфейсом пользователя.

## Программный код

#include <stdio.h>  
#include <limits.h>  
  
void inputMatrix(int matrix[50][50], int rows, int cols) {  
 printf("Введите элементы матрицы:\n");  
 for(int i=0; i<rows; i++)  
 for(int j=0; j<cols; j++)  
 scanf("%d", &matrix[i][j]);  
}  
  
int countNonZeroRows(int matrix[50][50], int rows, int cols) {  
 int count = 0;  
 for(int i=0; i<rows; i++) {  
 int zero\_found = 0;  
 for(int j=0; j<cols; j++)  
 if(matrix[i][j] == 0) zero\_found = 1;  
 if(!zero\_found) count++;  
 }  
 return count;  
}  
  
int findMaxRepeatingElement(int matrix[50][50], int rows, int cols) {  
 int freq[100] = {0}, max\_repeat = INT\_MIN;  
 for(int i=0; i<rows; i++)  
 for(int j=0; j<cols; j++)  
 freq[matrix[i][j] + 50]++;  
   
 for(int i=0; i<100; i++)  
 if(freq[i] > 1 && (i-50) > max\_repeat)  
 max\_repeat = i-50;  
  
 return max\_repeat;  
}  
  
int main() {  
 int matrix[50][50], rows, cols;  
 printf("Введите размер матрицы (rows cols): ");  
 scanf("%d%d", &rows, &cols);  
 inputMatrix(matrix, rows, cols);  
  
 printf("Строк без нулей: %d\n", countNonZeroRows(matrix, rows, cols));  
 int max\_repeat = findMaxRepeatingElement(matrix, rows, cols);  
 if(max\_repeat != INT\_MIN)  
 printf("Максимальный повторяющийся элемент: %d\n", max\_repeat);  
 else  
 printf("Повторяющиеся элементы отсутствуют.\n");  
  
 return 0;  
}

## Результаты вычисления

Изображение выглядит как текст, снимок экрана, Шрифт

Контент, сгенерированный ИИ, может содержать ошибки.