Министерство науки и высшего образования

Пензенский государственный университет

Кафедра “Вычислительная техника”

**Отчет**

по лабораторной работе №1

по курсу “ Логика и основы алгоритмизации в инженерных задачах”

на тему “Простые структуры данных”

Выполнили

студенты группы 22ВВП2:

Гавин В.Н.

Дулатов Д.А.

Приняли

Акифьев И.В.

Юрова О.В.

Пенза 2023

**Задание 1**: написать программу, вычисляющую разницу между максимальным и минимальным элементами массива.

**Задание 2**: написать программу, реализующую инициализацию массива случайными числами.

**Задание 3**: написать программу, реализующую создание массива произвольного размера, вводимого с клавиатуры.

**Задание 4**: написать программу, вычисляющую сумму значений в каждом столбце (или строке) двумерного массива.

**Задание 5:** написать программу, осуществляющую поиск среди структур student структуру с заданными параметрами (фамилией, именем и т.д.).

**Описание метода решения задачи**

**Задание 1:**

Создать массив чисел произвольной длины. Определить минимальное и максимальное значения в массиве. Для этого можно перебрать все элементы массива в цикле и сравнивать каждый элемент с текущими значениями min и max, обновляя их по мере прохождения цикла. Вычесть min из max, чтобы получить искомую разницу. Вывести результат - разницу max - min.

**Задание 2:**

Для инициализации массива случайными числами нужно использовать генератор псевдослучайных чисел. Пройти циклом по всем элементам массива и присвоить каждому элементу случайное число.

**Задание 3:**

Для инициализации массива случайными числами нужно использовать генератор псевдослучайных чисел. Пройти циклом по всем элементам массива и присвоить каждому элементу случайное число.

**Задание 4:**

Считать с клавиатуры размер массива N. Создать массив размером N элементов, выделяя память с помощью функции malloc(). Далее нужно пройти циклом по всем элементам массива и присвоить каждому элементу случайное число.

**Задание 5:**

Описать структуру Student. Создать массив структур Students из N элементов. Заполнить массив данными. Циклом пройти по всему массиву структур Students и для каждого элемента проверить совпадение значений полей с введенными данными для поиска. Вывести на экран найденный элемент или сообщение, что такого элемента нет.

**Листинг**

**Задание 1**:

**#include <stdio.h>**

**#include <locale.h>**

**int main() {**

**setlocale(LC\_ALL, "Rus");**

**int array[] = { 5, 8, 2, 10, 3 }; // Пример массива**

**int size = sizeof(array) / sizeof(array[0]); // Вычисляем размер массива**

**int max = array[0]; // Переменная для хранения максимального элемента**

**int min = array[0]; // Переменная для хранения минимального элемента**

**// Находим максимальный и минимальный элементы**

**for (int i = 1; i < size; i++) {**

**if (array[i] > max) {**

**max = array[i];**

**}**

**if (array[i] < min) {**

**min = array[i];**

**}**

**}**

**int difference = max - min; // Вычисляем разницу**

**printf("Исходный массив: ");**

**for (int i = 0; i < size; i++) {**

**printf("%d ", array[i]);**

**}**

**printf("\n");**

**printf("Разница между максимальным и минимальным элементами: %d\n", difference);**

**return 0;**

**}**

**Задание 2**:

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <time.h>**

**#include <locale.h>**

**int main() {**

**setlocale(LC\_ALL, "RUS");**

**int n = 10; // размер массива**

**// выделение памяти под массив**

**int \*arr = (int\*)malloc(n \* sizeof(int));**

**srand(time(NULL));**

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

**arr[i] = rand() % 101;**

**}**

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

**printf("%d ", arr[i]);**

**}**

**// освобождение памяти**

**free(arr);**

**return 0;**

**}**

**Задание 3**:

**#define \_CRT\_SECURE\_NO\_WARNINGS**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <time.h>**

**#include <locale.h>**

**int main() {**

**setlocale(LC\_ALL, "Russian");**

**int size;**

**printf("Введите размер массива: ");**

**scanf("%d", &size);**

**if (size <= 0) {**

**printf("Размер массива должен быть положительным числом.\n");**

**return 0;**

**}**

**int\* array = (int\*)malloc(size \* sizeof(int));**

**srand(time(NULL)); // Инициализация генератора случайных чисел**

**printf("Сгенерированный массив:\n");**

**for (int i = 0; i < size; i++) {**

**array[i] = rand() % 101; // Генерация случайного числа от 0 до 100**

**printf("%d ", array[i]);**

**}**

**printf("\n");**

**free(array);**

**return 0;**

**}**

**Задание 4**:

**#define \_CRT\_SECURE\_NO\_WARNINGS**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <locale.h>**

**int main() {**

**setlocale(LC\_ALL, "Russian");**

**int rows, cols;**

**printf("Введите количество строк: ");**

**scanf("%d", &rows);**

**printf("Введите количество столбцов: ");**

**scanf("%d", &cols);**

**int\*\* matrix = (int\*\*)malloc(rows \* sizeof(int\*));**

**for (int i = 0; i < rows; i++) {**

**matrix[i] = (int\*)malloc(cols \* sizeof(int));**

**}**

**// Заполнение матрицы случайными числами**

**for (int i = 0; i < rows; i++) {**

**for (int j = 0; j < cols; j++) {**

**matrix[i][j] = rand() % 10;**

**}**

**}**

**// Вывод матрицы на экран**

**for (int i = 0; i < rows; i++) {**

**for (int j = 0; j < cols; j++) {**

**printf("%d ", matrix[i][j]);**

**}**

**printf("\n");**

**}**

**for (int i = 0; i < rows; i++) {**

**int row\_sum = 0;**

**for (int j = 0; j < cols; j++) {**

**row\_sum += matrix[i][j];**

**}**

**printf("Сумма элементов в %d строке: %d\n", i+1, row\_sum);**

**}**

**// Освобождение памяти**

**for (int i = 0; i < rows; i++) {**

**free(matrix[i]);**

**}**

**free(matrix);**

**return 0;**

**}**

**Задание 5**:

**#define \_CRT\_SECURE\_NO\_WARNINGS**

**#include <stdio.h>**

**#include <string.h>**

**struct Student {**

**char firstName[50];**

**char lastName[50];**

**char faculty[50];**

**int studentID;**

**};**

**int main() {**

**struct Student students[] = {**

**{"John", "Doe", "Computer Science", 1001},**

**{"Jane", "Doe", "Physics", 1002},**

**{"Peter", "Smith", "Mathematics", 1003}**

**};**

**int n = sizeof(students) / sizeof(students[0]);**

**char firstName[50], lastName[50], faculty[50];**

**int id;**

**// Ввод параметров**

**printf("Enter first name: ");**

**fgets(firstName, 50, stdin);**

**firstName[strlen(firstName) - 1] = '\0';**

**printf("Enter last name: ");**

**fgets(lastName, 50, stdin);**

**lastName[strlen(lastName) - 1] = '\0';**

**printf("Enter faculty: ");**

**fgets(faculty, 50, stdin);**

**faculty[strlen(faculty) - 1] = '\0';**

**printf("Enter student ID: ");**

**scanf("%d", &id);**

**// Поиск**

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

**if (strlen(firstName) > 0 && strcmp(firstName, students[i].firstName) != 0) {**

**continue;**

**}**

**if (strlen(lastName) > 0 && strcmp(lastName, students[i].lastName) != 0) {**

**continue;**

**}**

**if (strlen(faculty) > 0 && strcmp(faculty, students[i].faculty) != 0) {**

**continue;**

**}**

**if (id != 0 && id != students[i].studentID) {**

**continue;**

**}**

**// Вывод результата**

**printf("Found: %s %s, Faculty: %s, ID: %d\n",**

**students[i].firstName, students[i].lastName,**

**students[i].faculty, students[i].studentID);**

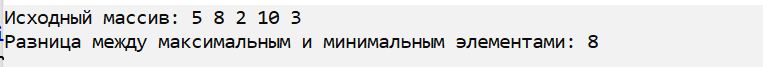
**}**

**return 0;**

**}**

**Результаты работы программы**

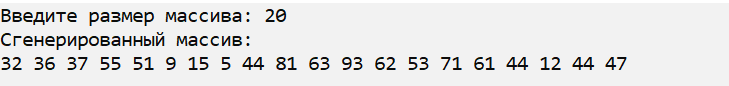
**Задание 1**:



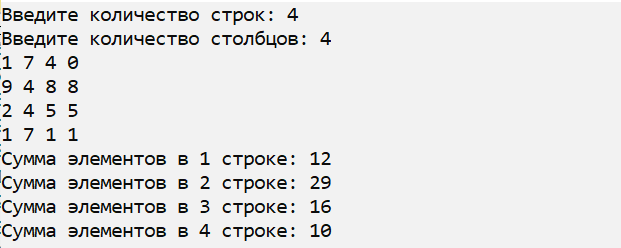
**Задание 2**:



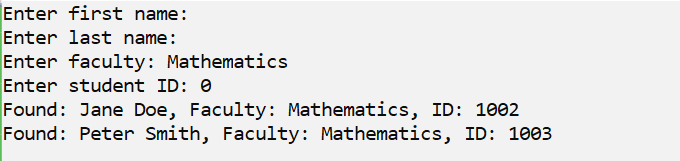
**Задание 3**:



**Задание 4**:



**Задание 5**:



**Вывод**

В ходе выполнения работы были закреплены навыки создания и инициализации одномерных и двумерных массивов, а также работы со структурами в языке программирования Си. Получен опыт разработки алгоритмов обработки данных с использованием базовых структур.

Работа позволила расширить практические знания и умения по программированию на языке Си.