

Министерство науки и высшего образования РФ
Пензенский государственный университет
Кафедра “Вычислительная техника”

Отчёт

по лабораторной работе №1
по курсу “Логика и основы алгоритмизации в инженерных задачах”
на тему “Простые структуры данных”

Выполнил студент гр. 22ВВВ3:
Кулахметов С.И.

Приняли:
к.т.н., доцент Юрова О.В.
к.э.н., доцент Акифьев И.В.

Пенза 2023

Цель работы

Ознакомиться с простыми структурами данных, такими как одномерные и двумерные массивы (в том числе и динамические) и простые структуры. В ходе лабораторной работы необходимо выполнить 5 заданий и ознакомиться с принципами работы простейших структур данных.

Лабораторное задание

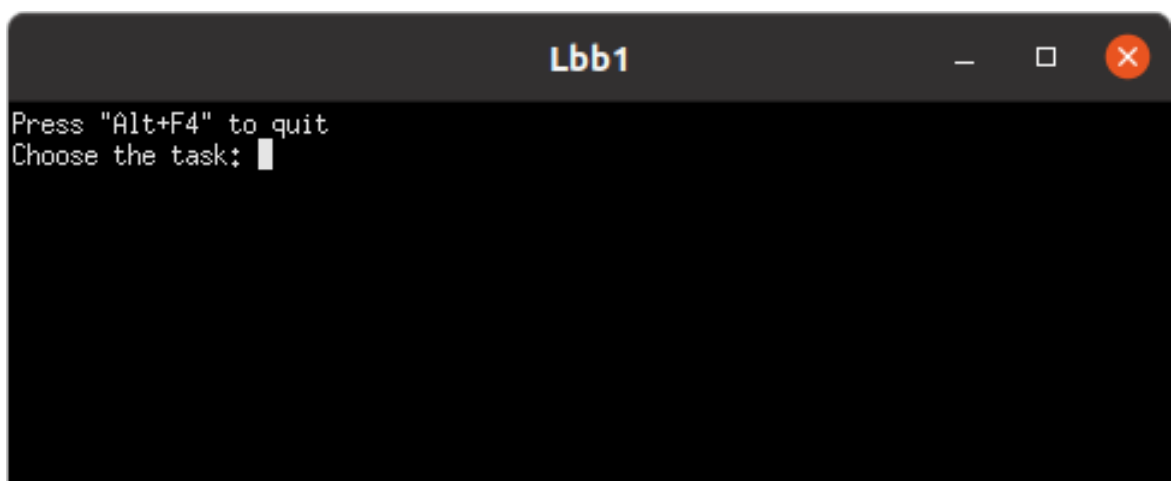
1. Написать программу, вычисляющую разницу между максимальным и минимальным элементами массива.
2. Написать программу, реализующую инициализацию массива случайными числами.
3. Написать программу, реализующую создание массива произвольного размера, вводимого с клавиатуры.
4. Написать программу, вычисляющую сумму значений в каждом столбце (или строке) двумерного массива.
5. Написать программу, осуществляющую поиск среди структур *student* структуру с заданными параметрами.

Пояснительный текст к программе

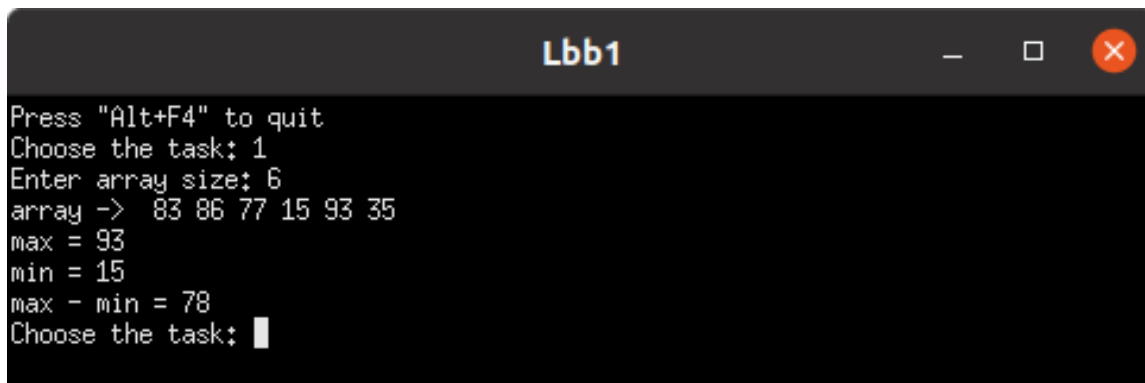
Программа написана на языке C++, имеет консольный интерфейс и базируется на 3-х основных функциях: *MaxMinArray()*, отвечающая за выполнение заданий №1-3; *SummArray()*, отвечающая на выполнение задания №4; и функция *StructSearch()*, в которой реализовано решение задания №5.

Результаты работы программы

При запуске программы в консоль выводится предложение о выборе задачи с 1-й по 5-ю.

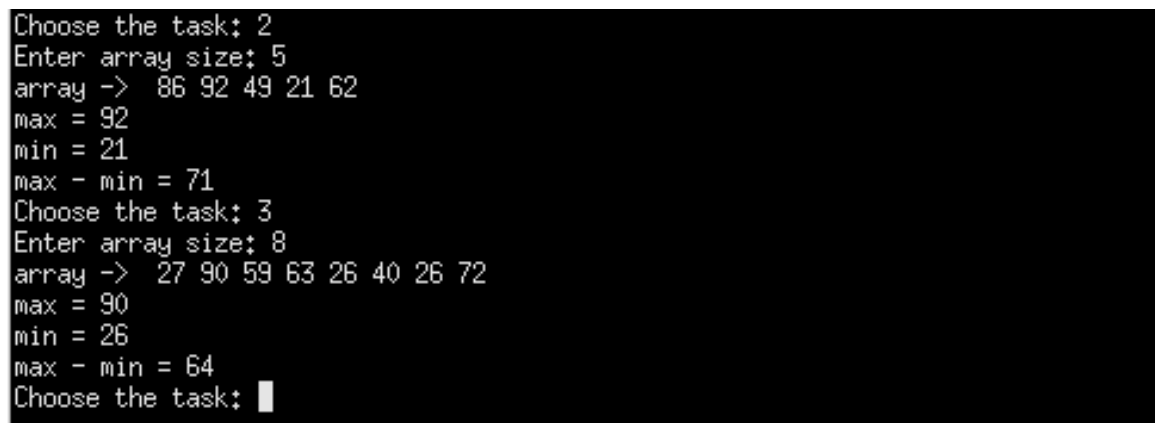


При выборе первой задачи выводится запрос на ввод количества элементов массива, массив заполняется псевдорандомными числами и находится разность максимального и минимального значений.



```
Lbb1
Press "Alt+F4" to quit
Choose the task: 1
Enter array size: 6
array -> 83 86 77 15 93 35
max = 93
min = 15
max - min = 78
Choose the task: █
```

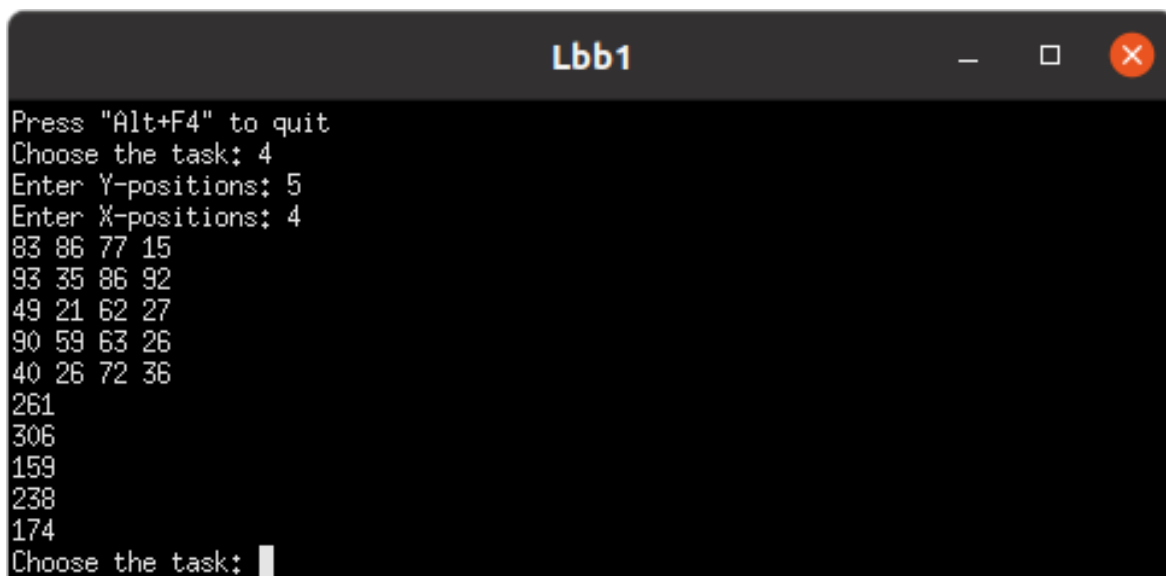
Аналогичная ситуация происходит при выборе 2-го и 3-го пункта, так как вызываемая функция совмещает в себе решения одновременно заданий 1-3 (рис. 1).



```
Lbb1
Choose the task: 2
Enter array size: 5
array -> 86 92 49 21 62
max = 92
min = 21
max - min = 71
Choose the task: 3
Enter array size: 8
array -> 27 90 59 63 26 40 26 72
max = 90
min = 26
max - min = 64
Choose the task: █
```

Рисунок 1 – решение заданий №2 и №3

При выборе задачи №4 нужно ввести количество строк и столбцов в двумерном массиве, после чего он будет заполнен псевдорандомными числами и мы получим суммы элементов каждой строки матрицы. Проверка правильности работы алгоритма представлена на иллюстрации 2.



```
Lbb1
Press "Alt+F4" to quit
Choose the task: 4
Enter Y-positions: 5
Enter X-positions: 4
83 86 77 15
93 35 86 92
49 21 62 27
90 59 63 26
40 26 72 36
261
306
159
238
174
Choose the task: █
```

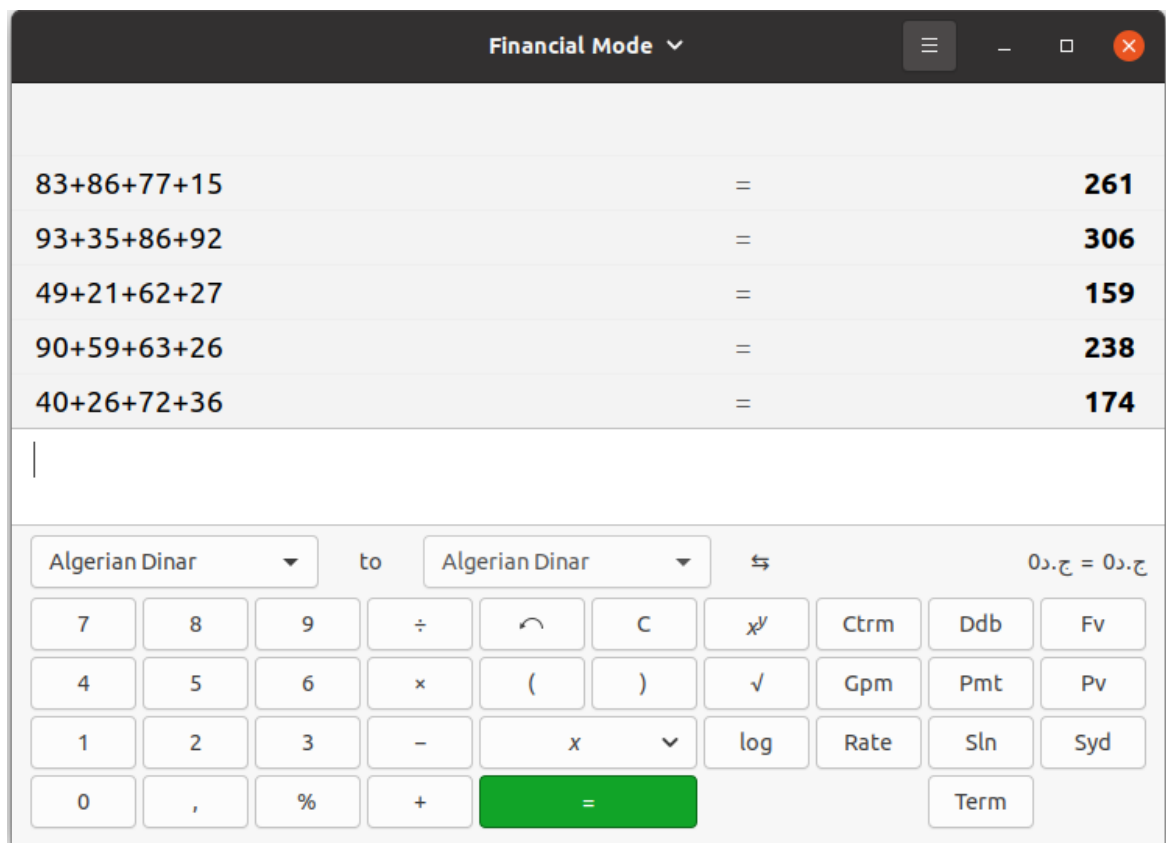


Рисунок 2 – Проверка правильности расчётов программы

Если ввести в консоль цифру 5, то это будет соответствовать выбору 5-го задания. После этого необходимо заполнить предложенные структуры требуемыми данными. Как только структуры будут заполнены выводится поле в которое можно ввести фамилиюю или имя искомого студента и вам будет выдана требуемая информация о нём.



Вывод

В ходе данной лабораторной работы я продемонстрировал применение простых структур данных, таких как одномерный/двумерный массив, строки и простейшую структуру. Рассмотрел принципы их работы.

Ссылка на *GitHub* репозиторий с лабораторной работой

<https://github.com/KulakhmetovS/Lab1/commits/main>

Приложение А

Листинг программы

```
#include <iostream>
#include <stdlib.h>
#include <string>

using namespace std;

void MaxMinArray(int); //Выполняет задания 1-3
void SummArray();    //Выполняет задание 4
void StructSearch(); //Выполняет задание 5

int main()
{
    int action = 0, array_size = 0;

    cout << "Press \"Alt+F4\" to quit" << endl;

    while(1)
    {
        cout << "Choose the task: ";
        cin >> action;
        if ((action < 1) || (action > 5)) {cout << "Invalid
operation" << endl;}
        else if(action == 1) MaxMinArray(array_size);
        else if(action == 2) MaxMinArray(array_size);
        else if(action == 3) MaxMinArray(array_size);
        else if(action == 4) SummArray();
        else if(action == 5) StructSearch();
    }

    return 0;
}

void MaxMinArray(int array_size)
{
    int max = 0, min = 0, i, j;

    cout << "Enter array size: ";
    cin >> array_size;

    int *array = new int[array_size];

    cout << "array -> ";

    for(i = 0; i < array_size; i++)
    {
        array[i] = rand() % 100;
        cout << array[i] << " ";
    }

    cout << endl;
```

```

    for(j = 0; j < array_size; j++)
    {
        if(max < array[j])
        {
            max = array[j];
        }
    }

    min = max;

    for(j = 0; j < array_size; j++)
    {
        if(min > array[j])
        {
            min = array[j];
        }
    }

    cout << "max = " << max << endl << "min = " << min << endl <<
    "max - min = " << max - min << endl;

    delete[] array;
}

void SummArray()
{
    int X = 0, Y = 0, i, j, sum = 0;
    int** matrix;

    cout << "Enter Y-positions: ";
    cin >> Y;
    cout << "Enter X-positions: ";
    cin >> X;

    matrix = new int*[Y];
    for(i = 0; i < Y; i++)
        matrix[i] = new int[X];

    for(i = 0; i < Y; i++)
    {
        for(j = 0; j < X; j++)
        {
            matrix[i][j] = rand() % 100;
            cout << matrix[i][j] << " ";
        }
        cout << endl;
    }

    for(i = 0; i < Y; i++)
    {
        for(j = 0; j < X; j++)
        {
            sum = sum + matrix[i][j];
        }
    }
}

```

```

        cout << sum << endl;
        sum = 0;
    }

    for(i = 0; i < Y; i++)
    {
        delete[] matrix[i];
    }
    delete[] matrix;
}

void StructSearch()
{
    int i, n = 3;
    string serch;
    struct students
    {
        int age;
        string name;
        string surname;
    } student[n];

    for(i = 0; i < n; i++)
    {
        cout << "Enter the name: ";
        cin >> student[i].name;
        cout << "Enter the surname: ";
        cin >> student[i].surname;
        cout << "Enter the age: ";
        cin >> student[i].age;
        cout << endl;
    }

    cout << "Enter a search term: ";
    cin >> serch;

    for(i = 0; i < n; i++)
    {
        if((serch.compare(student[i].name) == 0) ||
(serch.compare(student[i].surname) == 0))
        {
            cout << student[i].name << " ";
            cout << student[i].surname << " ";
            cout << student[i].age << endl;
        }
    }
}

```