

ОТЧЕТ Лабораторная №2-3  
ФИО: Голев Никита Владимирович  
Группа: СКБ252

1. Задание:

Написать программу «Анализ данных сенсора», которая работает с одномерным динамическим массивом. Программа должна считывать N измерений температуры, вычислять статистику (среднее, минимум, максимум) и создавать новый динамический массив, содержащий только те значения, которые превышают среднее арифметическое. Обязательное условие — использование арифметики указателей и ручное управление памятью.

2. Описание алгоритма:

Главная функция main()

- I. Запрашивает у пользователя количество измерений с проверкой на корректность.
- II. Выделяет память под динамический массив.
- III. Вызывает функции ввода, расчета статистики и фильтрации.
- IV. Выводит результаты на экран.
- V. Освобождает выделенную память.

Функция inputData()

- I. Заполняет массив данными, используя арифметику указателей \*(p+i).
- II. Реализует проверку ввода: числа должны быть в диапазоне от -50 до +50.

Функции average()

- I. average: Проходит по массиву через указатели, суммирует элементы и возвращает среднее.

Функции minValue(), maxValue()

- I. minValue / maxValue: Находят минимальное и максимальное значения в массиве соответственно.

Функция filterAboveAverage()

- I. Сначала подсчитывает количество элементов, превышающих среднее значение.
- II. Выделяет память под новый массив нужного размера.
- III. Заполняет новый массив подходящими значениями и возвращает указатель на него.

3. Код:

```
#include <iostream>
using namespace std;

void inputData(float* p, int size)
{
    for (int i = 0; i<size;i++)
    {
        while (true)
        {
            cin >>*(p+i);
            if (cin.fail()==true)
            {
                cin.clear();
                cin.ignore(252,'\'\n\'');
                cout<< "Число!!!";
                continue;
            }
            if (*(p + i) < -50 or *(p + i) > 50)
            {
                cout << "От -50 до 50!";
                cout << '\n';
                continue;
            }
            break;
        }
    }
}

float average(const float* p, int size)
{
    float sum =0;
    for (int i = 0; i <size; ++i)
    {
        sum += *(p + i);
    }
    return sum/size;
}

float maxValue(const float* p, int size)
```

```

{
    float max = *p;
    for (int i = 1; i <size; ++i)
    {
        if (*(p+i)>max)
        {
            max = *(p+i);
        }
    }
    return max;
}

float minValue(const float* p, int size)
{
    float min = *p;
    for (int i = 1; i <size; ++i)
    {
        if (*(p+i)<min)
        {
            min = *(p+i);
        }
    }
    return min;
}

float* filterAboveAverage(const float* p, int size, int& newCount)
{
    newCount = 0;
    float srznach = average(p,size);
    for (int i = 0; i <size; ++i)
    {
        if (*(p+i)>srznach)
        {
            newCount++;
        }
    }
    if (newCount == 0) return 0;
    float* result = new float[newCount];
    int c = 0;
    for (int i = 0; i <size; ++i)
    {
        if (*(p+i)>srznach)
        {
            *(result+c) = *(p+i);
            c++;
        }
    }
}

```

```
    }
    return result;
}
int main()
{
    cout << "==== Анализ сенсора ===\n";
    cout << "Введите количество измерений: ";
    int size;
    while (true)
    {
        if (!(cin >> size))
        {
            cin.clear();
            cin.ignore(252, '\n');
            cout << "Ошибка! Введите ЦЕЛОЕ положительное число: ";
            continue;
        }
        if (size <= 0)
        {
            cin.ignore(252, '\n');
            cout << "Количество измерений должно быть больше 0! Повторите ввод: ";
            continue;
        }
        cin.ignore(252, '\n');
        break;
    }
}
```

```
    float* data = new float[size];
```

```
    cout << "Введите значения: ";
    inputData(data, size);
```

```
    float srznach = average(data, size);
    float max = maxValue(data, size);
    float min = minValue(data, size);
```

```
    cout << "Среднее значение: " << srznach << '\n';
    cout << "Минимум: " << min << '\n';
    cout << "Максимум: " << max << '\n';
```

```
    int newCount;
    float* above = filterAboveAverage(data, size, newCount);
```

```
    cout << "Значения выше среднего:\n";
    for (int i = 0; i < newCount; i++)
    {
        cout << *(above + i) << " ";
    }
    cout << endl;
```

```
    delete[] data;
    delete[] above;
    return 0;
}
```

4. Ссылка на репозиторий:  
<https://github.com/nikgolich-ship-it/Nikita-Golev-SCB252.git>