Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Тульский государственный университет»

КАФЕДРА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

**ПРИМЕНЕНИЕ СТРУКТУР В ЯЗЫКЕ**

**ПРОГРАММИРОВАНИЯ СИ**

отчет о лабораторной работе № 5

по дисциплине

*ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ*

***ВАРИАНТ 13***

|  |  |  |
| --- | --- | --- |
| Выполнила: | ст. гр. 230711 | Павлова В.С. |
| Проверил: | асс. каф. ИБ | Курбаков М.Ю. |

Тула, 2023 г.

**ЦЕЛЬ И ЗАДАЧА РАБОТЫ**

**Цель:** изучить структурированный тип в языке C.

**Задача:** в данной работе требуется написать программу, демонстрирующую использование изученных принципов.

**ЗАДАНИЕ НА РАБОТУ**

Написать программу, реализующую описание структур и необходимых действий над ними, исходя из варианта задания.

№13. Создать список студентов в группе. Определить средний рост и вес студентов.

# **СХЕМА ПРОГРАММЫ**

Схема алгоритма программы, реализующей обработку пользовательской структуры данных, представлена на рисунке 1.



Рисунок 1 – Схема алгоритма программы, реализующей обработку пользовательской структуры данных

# **ТЕКСТ ПРОГРАММЫ**

Текст программы на языке программирования С/С++, реализующей обработку пользовательской структуры данных, представлен в листинге 1.

## **Листинг 1. Текст программы**

#include <iostream>

#include <vector>

#include <string>

struct Student

{

public:

std::string fullName;

## **Листинг 1. Текст программы (продолжение)**

double weightInKilograms;

double heightInMeters;

Student(std::string \_name = "", double weight = 0, double height = 0)

{

fullName = \_name;

weightInKilograms = weight;

heightInMeters = height;

}

~Student(){}

};

int main()

{

system("chcp 1251");

setlocale(LC\_ALL, "RUSSIAN");

int n;

double height, weight, averageHeight = 0, averageWeight = 0;

std::cout << "Введите количество студентов в группе: "; std::cin >> n;

std::vector<Student> studentsList;

std::string name;

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

{

std::cout << "Введите имя студента: ";

std::getline(std::cin >> std::ws, name);

std::cout << "Введите его рост (м) и вес (кг): ";

std::cin >> height >> weight;

Student person(name, weight, height);

studentsList.push\_back(person);

averageHeight += height;

averageWeight += weight;

}

std::cout << "\nСписок студентов в группе:\n";

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

{

std::cout << i + 1 << ". " << studentsList[i].fullName << ": "

<< studentsList[i].heightInMeters << " метров, "

<< studentsList[i].weightInKilograms << " килограмм.\n";

}

std::cout << "\nСредний рост " << averageHeight / n << ", средний вес " << averageWeight / n;

return 0;

}

**ИНСТРУКЦИЯ ПРОГРАММИСТА**

Структуры данных, используемые, в программе приведены в таблицах 1 и 2.

Таблица 1 – Структуры данных в основной программе

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Предназначение** |
| studentsList | Vector<Student> | Список объектов структуры Student |
| person | Student | Экземпляр структуры Student |

Таблица 1 – Структуры данных в основной программе (продолжение)

|  |  |  |
| --- | --- | --- |
| averageHeight | double | Средний рост в метрах |
| averageWeight | double | Вес в килограммах |
| height | double | Рост в метрах |
| weight | double | Вес в килограммах |
| n | int | Количество студентов в группе |

Таблица 2 – Описание разработанной структуры Student

|  |  |  |
| --- | --- | --- |
| struct Student | | |
| **Поля/свойства (элементы данных) структуры** | | |
| Название и тип | Описание | |
| string fullName | ФИО студента | |
| double weightInKilograms | Вес в килограммах | |
| double heightInMeters | Рост в метрах | |
| **Методы (функции-элементы) структуры** | | |
| Название и тип возвращаемого значения | Аргументы | Описание |
| Student() | string \_name = "", double weight = 0, double height = 0 | Конструктор структуры |
| ~Student() | void | Деструктор структуры |

# **ДЕМОНСТРАЦИОННЫЙ ПРИМЕР**

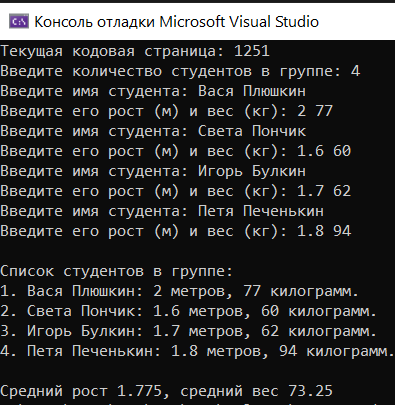
Пусть имеется следующий список из 4 студентов:

1. Света Пончик: 1.6 метров, 60 килограмм.
2. Игорь Булкин: 1.7 метров, 62 килограмм.
3. Вася Плюшкин: 2 метра, 77 килограмм.
4. Петя Печенькин: 1.8 метров, 94 килограмм.

Тогда их средний вес составляет кг.

Их средний рост составляет м.

Результат работы программы приведен на рисунке 2.

  
Рисунок 2 – Пример работы программы

# **ВЫВОДЫ**

В ходе выполнения работы я ознакомилась с таким типом данных, определяемых пользователем, как структура – совокупность переменных, объединенных одним именем, предоставляющая общепринятый способ совместного хранения информации. Объявление структуры приводит к образованию шаблона, используемого для создания объектов структуры. Переменные, образующие структуру, называются членами структуры. Структуры полезны, когда надо объединить несколько переменных с разными типами под одним именем.