МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГАОУ ВО «СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ ИНФОРМАЦИОННЫХ

ТЕХНОЛОГИЙ И ТЕЛЕКОММУНИКАЦИЙ

КАФЕДРА ПРИКЛАДНОЙ ИНФОРМАТИКИ

**Лабораторная работа №4**

по дисциплине

«Объектно-ориентированное программирование»

**Выполнил:**

Попов Антон Андреевич

Студент 2 курса группы ПИН-б-о-22-1

Направления подготовки

09.03.03 Прикладная информатика

очной формы обучения

Ставрополь, 2023 г.

**Тема:** Стандартные потоки

**Цель работы**: изучить стандартные потоки и научится реализовать их.

**Ход работы**

Вариант - 17

1. Определить класс с именем ZNAK, содержащий следующие поля: фамилия, имя; знак Зодиака; день рождения (массив из трех чисел). Определить методы доступа к этим полям и перегруженные операции извлечения и вставки для объектов типа ZNAK.
2. Написать программу, выполняющую следующие действия:

* ввод с клавиатуры данных в массив, состоящий из восьми объектов типа ZNAK; записи должны быть упорядочены по знакам Зодиака;
* вывод на экран информации о людях, родившихся в месяц, значение которого введено с клавиатуры;
* если таких нет, выдать на дисплей соответствующее сообщение.

Листинг приведён в файлах:

[main.cpp](https://github.com/Antrt/-/blob/main/LR4cpp/main.cpp) [ZNAK.h](https://github.com/Antrt/-/blob/main/LR4cpp/ZNAK.h) [ZNAK.cpp](https://github.com/Antrt/-/blob/main/LR4cpp/ZNAK.cpp) [ZNAKTest.h](https://github.com/EGP24/oop/blob/main/cppLR4/ZNAKTest.h) [ZNAKTest.cpp](https://github.com/Antrt/-/blob/main/LR4cpp/ZNAKTest.cpp)

Также приведена [UML-диаграмма](https://github.com/Antrt/-/blob/main/LR4cpp/uml.png) проекта

В заголовочном файле *ZNAK.h* определён класс *ZNAK*, в файле *ZNAK.cpp* приведена реализация класса *ZNAK*. В заголовочном файле *ZNAKTest.h* определены функции тестирования, в файле *ZNAKTest.cpp* приведены реализации функций тестирования. В файле *main.cpp* реализована функция *main*, тестирующая класс *ZNAK* ипредоставляющая интерфейс для работы с классом.

Ссылка на [репозиторий](https://github.com/Antrt/-/tree/main/LR4cpp), содержащий полностью выполненные задания.

**Вывод:** изучил стандартные потоки и научился реализовать их.