**Министерство науки и высшего образования Российской Федерации**

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**ITMO University**

**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5**

**По дисциплине** Объектно-ориентированное программирование

**Обучающийся** Зорина Яна Сергеевна

**Факультет** Факультет инфокоммуникационных технологий

**Группа** К3222

**Направление подготовки** 11.03.02 Инфокоммуникационные технологии и системы связи

**Образовательная программа** Программирование в инфокоммуникационных системах

**Обучающийся** 12.12.2023  Зорина Я.С.

(дата) (подпись) (Ф.И.О.)

**Руководитель**  Васильев С.Ю.

(дата) (подпись) (Ф.И.О.)

Санкт Петербург

2023

**СОДЕРЖАНИЕ**

[Введение 3](#_Toc153301848)

[ГЛАВА 1. Ход работы 4](#_Toc153301849)

[1.1 Упражнение 1 4](#_Toc153301850)

[1.2 Упражнение 2 5](#_Toc153301851)

[1.3 Упражнение 3 6](#_Toc153301852)

[1.4 Упражнение 4 7](#_Toc153301853)

[1.5 Упражнение 5 8](#_Toc153301854)

[1.6 Упражнение 6 9](#_Toc153301855)

[1.6 Упражнение 6 10](#_Toc153301856)

# Введение

Целью лабораторной работы является изучение наследования как важного элемента объектноориентированного программирования и приобретение навыков реализации иерархии классов..

Отчёт содержит одну главу, каждая подглава посвящена одному упражнению из лабораторной работы.

Для выполнения лабораторной работы использовалась программа «Visual Studio».

По этой [ссылке](https://github.com/moryacho/OOP_2023) доступен репозиторий, в котором каждой лабораторной работе отведена папка, в которых доступны программы для упражнений.

# ГЛАВА 1. Ход работы

## 1.1 Упражнение 1

Данная программа представляет собой пространство имен MyClass, которое содержит несколько классов для управления предметами и книгами в библиотеке.

Класс Item представляет базовый класс для управления предметами. Он содержит методы для проверки доступности предмета, получения инвентарного номера, выполнения операций "взять" и "вернуть", а также отображения состояния предмета.

Класс Book наследуется от класса Item и представляет собой модель книги в библиотеке. Он содержит дополнительные атрибуты, такие как автор, название, издательство, количество страниц, год издания, и стоимость аренды. Класс также содержит методы для установки данных книги, установки цены на аренду, отображения информации о книге, вычисления стоимости аренды и взятия книги.

Класс Program содержит точку входа в приложение. Программа создает экземпляр базового класса Item и отображает его состояние.

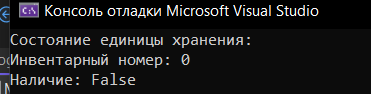
Результат выполнения программы представлен на рисунке 1.

Рисунок 1. Упражнение 1

## 1.2 Упражнение 2

Был добавлен класс Magazine, который наследуется от класса Item и представляет собой модель журнала в библиотеке. Он содержит атрибуты, такие как том, номер, название, год выпуска, а также наследует инвентарный номер и состояние доступности от базового класса. Класс также содержит методы для установки данных о журнале и отображения информации о нем.

В классе Program в методе Main создаются экземпляры предметов (объектов базового класса Item и объекта класса Magazine), устанавливаются и отображаются их свойства.

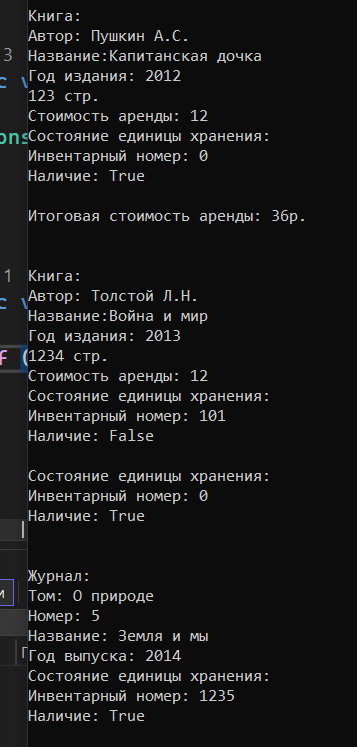
Результат выполнения программы представлен на рисунке 2.

Рисунок 2. Упражнение 2

## 1.3 Упражнение 3

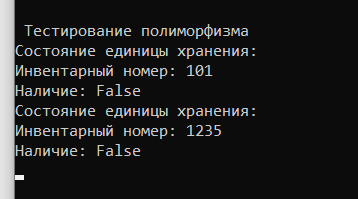
Был переопределён метод “Return” в классах “Magazine” и “Book”. Перед этим метод в классе-родителе “Item” был обозначен виртуальным. Каждый производный класс имеет свою версию ритуального метода. Результат выполнения этой части представлен на рисунке 3.

Рисунок 3. Упражнение 3

## 1.4 Упражнение 4

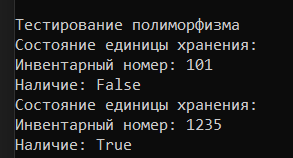
 Класс Item стал абстрактным, как и его метод Return. Результат выполнения части программы, показывающей это, представлен на рисунке 4.

Рисунок 4. Упражнение 4

## 1.5 Упражнение 5

Данный программный код создает модели точек и отрезков на плоскости, а также демонстрирует их свойства и функциональность.

Класс Point представляет собой модель точки с координатами x и y. Он содержит конструктор для установки координат точки, метод для отображения информации о точке, метод для вычисления расстояния между этой и другой точкой, а также переопределение метода ToString для возвращения строкового представления точки.

Класс Line представляет собой модель отрезка, который определяется двумя точками - начальной и конечной. Он содержит конструктор для установки начальной и конечной точек отрезка, метод для отображения информации об отрезке, а также метод для вычисления длины отрезка.

В методе Main класса Program создаются две точки p1 и p2 с координатами (0, 0) и (12, 13) соответственно. Затем создается отрезок line с использованием этих точек. После этого программа отображает информацию о точках и отрезке, а также вычисляет и отображает длину этого отрезка.

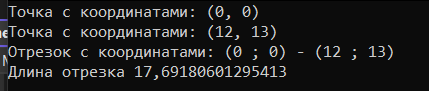
Результат работы программы представлен на рисунке 5.

Рисунок 5. Упражнение 5

## 1.6 Упражнение 6

Класс Dice представляет собой кость, которая может генерировать случайное число от 1 до 6. Он содержит конструктор, который инициализирует объект класса Random для генерации случайных чисел, а также метод random(), который возвращает случайное число от 1 до 6.

Класс Gamer представляет игрока в игре. У игрока есть имя и игровая сессия, которая представлена объектом класса Dice. Конструктор класса Gamer инициализирует имя игрока и создает новую игровую сессию с использованием объекта Dice. Класс также содержит переопределение метода ToString() для возвращения имени игрока и метод SessionGame() для генерации случайного числа от 1 до 6 в рамках игровой сессии.

В методе Main класса Program создается объект игрока g1 с именем "Niko". Затем в цикле генерируются числа от 1 до 6 с помощью метода SessionGame() объекта g1, и выводится информация о количестве очков, выпавших игроку "Niko" в каждом броске.

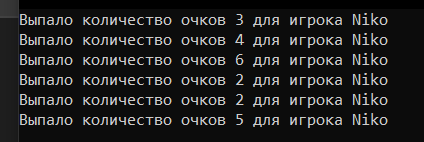
Результат выполнения программы представлен на рисунке 6.

Рисунок 6. Упражнение 6

## 1.6 Упражнение 6

Создаются следующие классы: абстрактный класс Progression, описывающий прогрессии, и два производных класса: ArithmeticProgression для арифметической прогрессии и GeometricProgression для геометрической прогрессии. В каждом из производных классов переопределен абстрактный метод GetElement, который возвращает элемент прогрессии по переданному параметру k.

Класс Program используется для тестирования работы этих прогрессий. В методе Main создаются объекты арифметической и геометрической прогрессии с заданными начальными значениями и шагами (разностями или знаменателями). Затем программа получает элементы прогрессий для заданных параметров k и выводит их значения на экран.

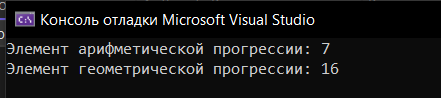
Результат выполнения программы представлен на рисунке 7.

Рисунок 7. Упражнение 7