**Лабораторная работа №12. Часть 1**

*«Классы. Наследование»*

**Рекомендации по программированию**

* Поля обязательно делать private.
* Интерфейс должен быть полным, т.е. предоставлять возможность выполнять любые разумные действия с классом. И одновременно минимально необходимым, т.е. без дублирования и пересечения возможностей методов.
* Каждый метод класса должен решать только одну задачу. Не стоит объединять два коротких независимых фрагмента кода в один метод.
* Если действия встречаются хотя бы дважды, стоит оформить их в отдельные функции.
* Все заданные методы, поля и экземпляры классов должны иметь осмысленные имена.

**Требования к лабораторной**

* Каждый разрабатываемый класс должен, как правило, содержать следующие элементы: поля с различными спецификаторами, конструкторы с параметрами и без параметров, методы, свойства.
* Методы и свойства должны обеспечивать непротиворечивый, полный, минимальный и удобный интерфейс класса.
* К каждому отчету, помимо формулировки задания и скрина консоли, нужно прикрепить еще диаграмму классов

**Задания для самостоятельного решения**

При выполнении данной работы необходимо определить базовый класс и производные от него классы. Предусмотреть передачу аргументов конструкторам базового класса; использование виртуальных и перегруженных функций; обработку исключительных ситуаций.

**Варианты с 1-10**

В следующих заданиях требуется создать базовый класс (как вариант абстрактный базовый класс) и определить свойства get ( ), set ( ), метод show ( ) и другие, специфические для данного класса. Создать производные классы, в которые добавить методы согласно индивидуальному заданию.

Часть методов переопределить. Создать массив объектов базового класса и заполнить объектами производных классов. Объекты произ­водных классов идентифицировать конструктором.

Вызвать метод show ( ) базового класса и просмотреть массив объектов.

1. Создать базовый класс «Транспортное средство» и производные классы «Автомобиль», «Велосипед», «Повозка». Подсчитать время и стоимость пере­возки пассажиров и грузов каждым транспортным средством.

2. Создать базовый класс «Грузоперевозчик» и производные классы «Самолет», «Поезд», «Автомобиль». Определить время и стоимость перевозки для указанных городов и расстояний.

3. Создать аналогичный базовый класс «Пассажироперевозчик» и производные классы «Самолет», «Поезд», «Автомобиль». Определить время и стоимость передвижения.

4. Создать базовый класс «Учащийся» и производные классы «Школьник» и «Студент». Создать массив объектов базового класса и заполнить этот массив объектами. Показать отдельно студентов и школьников.

5. Создать базовый класс «Музыкальный инструмент» и производные классы «Ударный», «Струнный», «Духовой». Создать массив объектов «Оркестр». Выдать состав оркестра, переопределив метод.

6. Создать абстрактный класс «Работник фирмы» и производные классы «Менеджер», «Администратор», «Программист». Создать массив объектов базового класса и заполнить этот массив объектами. Показать отдельно элементы каждого класса. Определить среднюю зп всех сотрудников.

7. Определить базовый класс «Множество» и производный класс «Кольцо» (операции сложения и умножения обе коммутативные и ассоциативные, связанные законом дистрибутивности; сложение обладает обратной операцией – вычитанием). Ввести кольца целых чисел, многочленов, систему классов целых чисел, сравнимых по модулю. Кольцо является полем, если в нем определена операция деления, кроме деления на нуль. Рациональные числа, дробно рациональные функции.

8. Создать абстрактный класс «Магазин» с производным классом «Стеллаж», от которого образовывается еще один класс «Полка». Создать массив объектов базового класса и заполнить этот массив объектами. Показать отдельно элементы каждого класса. Определить количество товаров на каждом стелаже.

9. Создать базовый класс «Домашнее животное» и производные классы «Собака», «Кошка», «Попугай» и др. С помощью конструктора установить имя каждого животного и его характеристики.

10. Создать базовый класс «Садовое дерево» и производные классы «Яблоня», «Вишня», «Груша» и др. Создать массив объектов каждого класса. Принять решение о пересадке каждого дерева в зависимости от возраста и плодоношения.

**Варианты 11-27**

1. Создать класс Item (единица хранения в библиотеке), содержащий данные-члены: invNumber – инвентарный номер и taken – взято на руки или имеется в наличии, а также методы:

virtual void Show(); //показать информацию о единице хранения

bool isAvailable(); // есть ли единица хранения в наличии ?

int GetinvNumber(); //возвращает инвентарный номер

void Take(); // операция «взять»

void Return(); // операция «вернуть»

Построить производные классы Book и Magazin. Класс Book содержит данные-члены: author, title, publisher, year и методы: Author();

Title(); Publisher(); YearOf Publishing(); Show().

Класс Magazin включает данные-члены: volume; number; year; title и методы: Volume(); Title(); Number(); Year(); Show().

abstract class Car

{

string mark;

protected int doors;

protected double price;

protected double speed;

protected double probeg;

protected int bakVolum;

public double Price { get { return price; } }

public Car(double price, double speed, double probeg, string mark, int doors, int bakVolum)

{

this.mark = mark;

this.price = price;

this.probeg = probeg;

this.speed = speed;

this.doors = doors;

this.bakVolum = bakVolum;

}

public Car(double price, string mark, int doors)

{

this.mark = mark;

this.price = price;

this.probeg = 0;

this.speed = 0;

this.doors = doors;

}

public Car()

{

this.mark = "";

this.price = 0;

this.probeg = 0;

this.speed = 0;

this.doors = 0;

}

public Car(double price)

{

this.mark = "";

this.price = price;

this.probeg = 0;

this.speed = 0;

this.doors = 0;

}

public void InputInfo()

{

this.mark = Console.ReadLine();

this.price = Convert.ToDouble(Console.ReadLine());

this.probeg = Convert.ToDouble(Console.ReadLine());

this.speed = Convert.ToDouble(Console.ReadLine());

this.doors = Convert.ToInt32(Console.ReadLine());

this.bakVolum = Convert.ToInt32(Console.ReadLine());

}

public double Time(int range)

{

return range / speed \* 60;

}

public double Fuel(int range, int fuel)

{

return range / fuel;

}

public abstract double NewProbeg(int range);

virtual public int Maslo(int range)

{

return range / 3000;

}

public int BakVolume(int range)

{

if (bakVolum==0)

{

return 0;

}

return range / bakVolum;

}

public int Omivaika(int range, int omivayka)

{

return range / omivayka;

}

public void ShowInfo()

{

Console.WriteLine(price + " " + speed + " " + probeg + " " + mark + " " + doors + " " + bakVolum + " " + price);

}

}

