|  |  |  |
| --- | --- | --- |
| **РАССМОТРЕНО И ОДОБРЕНО**  на заседании ЦМК  дисциплин профессионального цикла  Протокол № \_\_\_ от «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_20\_\_\_ г.  Председатель ЦМК \_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Е.Г.Белоус/ |  | **УТВЕРЖДАЮ**  Зам. директора по УР  \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Л.И.Гончарук  «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_\_ г. |

**Перечень задач для подготовки к промежуточной аттестации (экзамену)**

**Задание 1.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс с именем Rectangle.

В теле класса создать два поля, описывающие длины сторон double side1, side2.

Создать пользовательский конструктор Rectangle(double side1, double side2), в теле которого поля side1 и side2 инициализируются значениями аргументов.

Создать два метода, вычисляющие площадь прямоугольника - double AreaCalculator() и периметр прямоугольника - double PerimeterCalculator().

Создать два свойства double Area и double Perimeter с одним методом доступа get.

Написать программу, которая принимает от пользователя длины двух сторон прямоугольника и выводит на экран периметр и площадь.

**Задание 2.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс Book. Создать классы Title, Author и Content, каждый из которых должен содержать одно строковое поле и метод void Show().

Реализуйте возможность добавления в книгу названия книги, имени автора и содержания.

Выведите на экран разными цветами при помощи метода Show() название книги, имя автора и содержание.

**Задание 3.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать классы Point и Figure.

Класс Point должен содержать два целочисленных поля и одно строковое поле.

Создать три свойства с одним методом доступа get.

Создать пользовательский конструктор, в теле которого проинициализируйте поля значениями аргументов.

Класс Figure должен содержать конструкторы, которые принимают от 3-х до 5-ти аргументов типа Point.

Создать два метода: double LengthSide(Point A, Point B), который рассчитывает длину стороны многоугольника; void PerimeterCalculator(), который рассчитывает периметр многоугольника.

Написать программу, которая выводит на экран название и периметр многоугольника.

**Задание 4.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс с именем Address.

В теле класса требуется создать поля: index, country, city, street, house, apartment.

Для каждого поля, создать свойство с двумя методами доступа.

Создать экземпляр класса Address.

В поля экземпляра записать информацию о почтовом адресе.

Выведите на экран значения полей, описывающих адрес.

**Задание 5.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс Converter.

В теле класса создать пользовательский конструктор, который принимает три вещественных аргумента, и инициализирует поля соответствующие курсу 3-х основных валют, по отношению к BYN – public Converter(double usd, double eur, double rub).

Написать программу, которая будет выполнять конвертацию из BYN в одну из указанных валют, также программа должна производить конвертацию из указанных валют в BYN.

**Задание 6.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс Employee.

В теле класса создать пользовательский конструктор, который принимает два строковых аргумента, и инициализирует поля, соответствующие фамилии и имени сотрудника.

Создать метод рассчитывающий оклад сотрудника (в зависимости от должности и стажа) и налоговый сбор.

Написать программу, которая выводит на экран информацию о сотруднике (фамилия, имя, должность), оклад и налоговый сбор.

**Задание 7.** Используя Visual Studio, создайте проект по шаблону Console Application

Требуется: Создать класс Invoice.

В теле класса создать три поля int account, string customer, string provider, которые должны быть проинициализированы один раз (при создании экземпляра данного класса) без возможности их дальнейшего изменения.

В теле класса создать два закрытых поля string article, int quantity

Создать метод расчета стоимости заказа с НДС и без НДС.

Написать программу, которая выводит на экран сумму оплаты заказанного товара с НДС или без НДС.

**Задание 8.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс User, содержащий информацию о пользователе (логин, имя, фамилия, возраст, дата заполнения анкеты).

Поле дата заполнения анкеты должно быть проинициализировано только один раз (при создании экземпляра данного класса) без возможности его дальнейшего изменения.

Реализуйте вывод на экран информации о пользователе.

**Задание 9.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс, представляющий учебный класс ClassRoom.

Создайте класс ученик Pupil.

В теле класса создайте методы void Study(), void Read(), void Write(), void Relax().

Создайте 3 производных класса ExcelentPupil, GoodPupil, BadPupil от класса базового класса Pupil и переопределите каждый из методов, в зависимости от успеваемости ученика.

Конструктор класса ClassRoom принимает аргументы типа Pupil, класс должен состоять из 4 учеников.

Предусмотрите возможность того, что пользователь может передать 2 или 3 аргумента.

Выведите информацию о том, как все ученики экземпляра класса ClassRoom умеют учиться, читать, писать, отдыхать.

**Задание 10.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать класс Vehicle.

В теле класса создайте поля: координаты и параметры средств передвижения (цена, скорость, год выпуска).

Создайте 3 производных класса Plane, Саг и Ship.

Для класса Plane должна быть определена высота и количество пассажиров.

Для класса Ship — количество пассажиров и порт приписки.

Написать программу, которая выводит на экран информацию о каждом средстве передвижения.

**Задание 11.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создайте класс DocumentWorker.

В теле класса создайте три метода OpenDocument(), EditDocument(), SaveDocument().

Тело каждого из методов добавьте вывод на экран соответствующих строк: "Документ открыт", "Редактирование документа доступно в версии Про", "Сохранение документа доступно в версии Про".

Создайте производный класс ProDocumentWorker.

Переопределите соответствующие методы, при переопределении методов выводите следующие строки: "Документ отредактирован", "Документ сохранен в старом формате, сохранение в остальных форматах доступно в версии Эксперт".

Создайте производный класс ExpertDocumentWorker от базового класса ProDocumentWorker.

Переопределите соответствующий метод. При вызове данного метода необходимо выводить на экран "Документ сохранен в новом формате".

В теле метода Main() реализуйте возможность приема от пользователя номера ключа доступа pro и exp.

Если пользователь не вводит ключ, он может пользоваться только бесплатной версией (создается экземпляр базового класса), если пользователь ввел номера ключа доступа pro и exp, то должен создаться экземпляр соответствующей версии класса, приведенный к базовому – DocumentWorker.

**Задание 12.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создайте класс Printer.

В теле класса создайте метод void Print(string value), который выводит на экран значение аргумента.

Реализуйте возможность того, чтобы в случае наследования от данного класса других классов, и вызове соответствующего метода их экземпляра, строки, переданные в качестве аргументов методов, выводились разными цветами.

Обязательно используйте приведение типов.

**Задание 13.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создайте класс AbstractHandler.

В теле класса создать методы void Open(), void Create(), void Chenge(), void Save().

Создать производные классы XMLHandler, TXTHandler, DOCHandler от базового класса AbstractHandler.

Написать программу, которая будет выполнять определение документа и для каждого формата должны быть методы открытия, создания, редактирования, сохранения определенного формата документа.

**Задание 14.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создайте 2 интерфейса IPlayable и IRecodable.

В каждом из интерфейсов создайте по 3 метода void Play() / void Pause() / void Stop() и void Record() / void Pause() / void Stop() соответственно.

Создайте производный класс Player от базовых интерфейсов IPlayable и IRecodable.

Написать программу, которая выполняет проигрывание и запись.

**Задание 15.** Используя Visual Studio, создайте проект по шаблону Console Application.

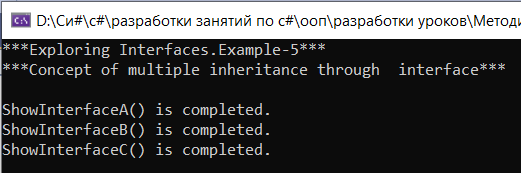
1. Создайте интерфейсы с именами InterfaceA, InterfaceB.

2. В каждом интерфейсе создайте соответствующие методы с именем ShowInterfaceA(), void ShowInterfaceB().

3. Создайте интерфейс с именем InterfaceC, реализующий множественное наследование.

4. Создайте класс MyClass, в котором реализуется интерфейс InterfaceС.

5. В методе Main() создайте объект myClassOb и вызовите все методы, чтобы в консольном окне был выведен следующий текст:

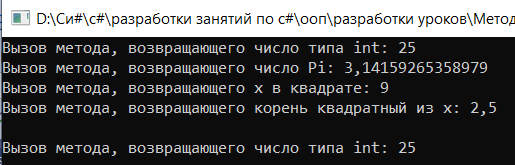


**Задание 16.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется объявить один интерфейс, в котором содержатся 4 метода:

1. метод, возвращающий число PI;
2. метод, возвращающий число типа int;
3. метод, возвращающий x в квадрате;
4. метод, возвращающий корень квадратный из x.

Реализовать интерфейс в классе MyClass, использующем этот интерфейс. Класс обязательно должен реализовать все методы интерфейса.



**Задание 17.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте интерфейс ISwitchable, в котором объявите два метода – включение и выключение.
2. Реализуйте интерфейс в классе TVSet с методами включения и выключения телевизора.
3. Реализуйте интерфейс в классе PersonalComputer с методами включения и выключения компьютера.
4. С использованием списка вызовите методы созданных классов.

**Задание 18.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте интерфейс ISwitchable, в котором объявите два метода – включение и выключение.
2. Реализуйте интерфейс в классе TVSet с методами включения и выключения телевизора.
3. Реализуйте интерфейс в классе PersonalComputer с методами включения и выключения компьютера.
4. С использованием списка вызовите методы созданных классов.

**Задание 19.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется:

Создать класс Article, содержащий следующие закрытые поля: название товара; название магазина, в котором продается товар; стоимость товара в BYN.

Создать класс Store, содержащий закрытый массив элементов типа Article.

Обеспечить следующие возможности: вывод информации о товаре по номеру с помощью индекса; вывод на экран информации о товаре, название которого введено с клавиатуры, если таких товаров нет, выдать соответствующее сообщение;

Написать программу, вывода на экран информацию о товаре.

**Задание 20.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Создать статический класс Calculator, с методами для выполнения основных арифметических операций.

Написать программу, которая выводит на экран основные арифметические операции.

**Задание 21.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Описать структуру с именем Train, содержащую следующие поля: название пункта назначения, номер поезда, время отправления.

Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в массив, состоящий из восьми элементов типа Train (записи должны быть упорядочены по номерам поездов); вывод на экран информации о поезде, номер которого введен с клавиатуры (если таких поездов нет, вывести соответствующее сообщение).

**Задание 22.** Создайте класс MyClass и структуру MyStruct, которые содержат в себе поля public string change.

В классе Program создайте два метода:

- static void ClassTaker(MyClass myClass), который присваивает полю change экземпляра myClass значение «изменено».

- static void StruktTaker(MyStruct myStruct), который присваивает полю change экземпляра myStruct значение «изменено».

Продемонстрируйте разницу в использовании классов и структур, создав в методе Main() экземпляры структуры и класса. Измените, значения полей экземпляров на «не изменено», а затем вызовите методы ClassTaker и StruktTaker. Выведите на экран значения полей экземпляров.

Проанализируйте полученные результаты.

**Задание 23.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте структуру с именем - Notebook.

Поля структуры: модель, производитель, цена.

В структуре должен быть реализован конструктор для инициализации полей и метод для вывода содержимого полей на экран.

**Задание 24.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте статический класс с методом void Print (string stroka, int color), который выводит на экран строку заданным цветом.

Используя перечисление, создайте набор цветов, доступных пользователю.

Ввод строки и выбор цвета предоставьте пользователю.

**Задание 25.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте перечисление, в котором будут содержаться должности сотрудников как имена констант.

Присвойте каждой константе значение, задающее количество часов, которые должен отработать сотрудник за месяц.

Создайте класс Accauntant с методом bool AskForBonus(Post worker, int hours), отражающее давать или нет сотруднику премию.

Если сотрудник отработал больше положеных часов в месяц, то ему положена премия.

**Задание 26.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте четыре лямбда оператора для выполнения арифметических действий: (Add – сложение, Sub – вычитание, Mul – умножение, Div – деление).

Каждый лямбда оператор должен принимать два аргумента и возвращать результат вычисления.

Лямбда оператор деления должен делать проверку деления на ноль.

Написать программу, которая будет выполнять арифметические действия, указанные пользователем.

**Задание 27.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте класс MyList<T>.

Реализуйте в простейшем приближении возможность использования его экземпляра аналогично экземпляру класса List<T>.

Минимально требуемый интерфейс взаимодействия с экземпляром, должен включать метод добавления элемента, индексатор для получения значения элемента по указанному индексу и свойство только для чтения для получения общего количества элементов.

**Задание 28.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте класс MyClass<T>, содержащий статический фабричный метод – T FacrotyMethod(), который будет порождать экземпляры типа, указанного в качестве параметра типа (указателя места заполнения типом – Т).

**Задание 29.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте класс CarCollection<T>.

Реализуйте в простейшем приближении возможность использования его экземпляра для создания парка машин.

Минимально требуемый интерфейс взаимодействия с экземпляром, должен включать метод добавления машин с названием машины и года ее выпуска, индексатор для получения значения элемента по указанному индексу и свойство только для чтения для получения общего количества элементов.

Создайте метод удаления всех машин автопарка.

**Задание 30.** Требуется: Описать структуру с именем Worker, содержащую следующие поля: фамилия и инициалы работника; название занимаемой должности; год поступления на работу.

Написать программу, выполняющую следующие действия: ввод с клавиатуры данных в массив, состоящий из пяти элементов типа Worker (записи должны быть упорядочены по алфавиту);

1. если значение года введено не в соответствующем формате выдает исключение.

2. вывод на экран фамилии работника, стаж работы которого превышает введенное значение.

**Задание 31.** Используя Visual Studio, создайте проект по шаблону Console Application.

Требуется: Описать структуру с именем Price, содержащую следующие поля:

1. название товара;

2. название магазина, в котором продается товар;

3. стоимость товара в BYN.

Написать программу, выполняющую следующие действия:

1. ввод с клавиатуры данных в массив, состоящий из двух элементов типа Price (записи должны быть упорядочены в алфавитном порядке по названиям магазинов);

2. вывод на экран информации о товарах, продающихся в магазине, название которого введено с клавиатуры (если такого магазина нет, вывести исключение).

**Задание 32.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте класс Calculator.

В теле класса создайте четыре метода для арифметических действий: (Add – сложение, Sub – вычитание, Mul – умножение, Div – деление).

Метод деления должен делать проверку деления на ноль, если проверка не проходит, сгенерировать исключение.

Пользователь вводит значения, над которыми хочет произвести операцию и выбрать саму операцию.

При возникновении ошибок должны выбрасываться исключения.

**Задание 33.** Используя Visual Studio, создайте проект по шаблону Console Application.

Представьте, что вы пишите приложение для Автостанции и вам необходимо создать простую коллекцию автомобилей со следующими данными:

1. марка автомобиля;

2. модель;

3. год выпуска;

4. цвет.

А также вторую коллекцию с моделью автомобиля, именем покупателя и его номером телефона.

Используя простейший LINQ запрос, выведите на экран информацию о покупателе одного из автомобилей и полную характеристику приобретенной ними модели.

**Задание 34.** Используя Visual Studio, создайте проект по шаблону Console Application.

Реализуйте один интерфейс с одним методом в производном классе.

1. Создайте интерфейс с именем IInterface.
2. Создайте в интерфейсе метод Method().
3. Создайте производный класс DerivedClass, который реализует интерфейс IInterface.
4. Реализуйте метод Method() в классе DerivedClass, чтобы в консольном окне выводился текст:

Метод, реализованный в классе DerivedClass

1. В методе Main() создайте объект dClass и вызовите метод Method().

**Задание 35.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте интерфейсы IInterface1, IInterface2, IInterface3.
2. Интерфейс IInterface2 наследуйте от интерфейса IInterface1.
3. Интерфейс IInterface3 наследуйте от интерфейса IInterface2.
4. В каждом интерфейсе создайте по одному методу с соответствующими именами: Method1(), Method2(), Method3().
5. Создайте производный класс DerivedClass, который наследуется от интерфейса IInterface3.
6. В классе DerivedClass реализуйте все три метода Method1(), Method2(), Method3().
7. В методе Main() создайте объект dClass и вызовите все методы, чтобы в консольном окне был выведен следующий текст:

Метод Method1

Метод Method2

Метод Method3

**Задание 36.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте класс Person со свойствами имени и возраста человека, реализующий интерфейс IComparable.

Создайте список и множество из четырех людей.

Отсортируйте объекты по имени в алфавитном порядке.

**Задание 37.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте абстрактный класс Человек, в котором будет свойство Имя и абстрактный метод СказатьПриветствие(), от этого класса будет несколько наследников, которые представляют национальность (русский, украинец, американец...). Должно получиться так, что при вызове метода СказатьПриветствие() выводилось приветствие на языке соответствующему национальности (Привет, Merhaba, Hi...).

**Задание 38.** Используя Visual Studio, создайте проект по шаблону Console Application.

Разработать класс для объекта Student. Включить в класс методы set (…), get (…), show (…). Определить другие методы. Student: Фамилия, Имя, Отчество, Дата рождения, Адрес, Телефон, Факультет, Курс. Создать массив объектов. Вывести: а) список студентов заданного факультета; б) списки студентов для каждого факультета и курса; в) список студентов, родившихся после заданного года.

**Задание 39.** Используя Visual Studio, создайте проект по шаблону Console Application.

Разработать класс для объекта Product: Наименование, Производитель, Цена, Срок хранения, Количество. Создать массив объектов. Вывести: а) список товаров для заданного наименования; б) список товаров для заданного наименования, цена которых не превышает указанной; в) список товаров, срок хранения которых больше заданного.

**Задание 40.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создается класс Computer с полями: модель model, объем ОЗУ ram и жесткого диска hdd. Поля инкапсулируются с ограничениями (объем ОЗУ от 2 до 32 Гбайт, жесткого диска от 200 до 2000 Гбайт). Вычисляется стоимость указанной модели и общая стоимость всех моделей.

**Задание 41.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте статический метод double Average (int m, int n), который вычисляет и возвращает среднее арифметическое всех целых чисел от *m* до *n* включительно.
2. Объявите тип делегата, соответствующий этому методу.
3. Объявите сам делегат.
4. Применить в программе созданный делегат 3 раза.

**Задание 42.** Используя Visual Studio, создайте проект по шаблону Console Application.

Реализовать обобщенный класс Point, который определяет точку на координатной плоскости. В классе реализовать:

* внутренние поля x, y;
* конструктор с 2 параметрами;
* свойства доступа к внутренним полям класса;
* метод, выводящий значения внутренних полей класса.

Написать программу, иллюстрирующую работу данного класса. В качестве универсальных параметров используйте типы int, uint и double.

**Задание 43.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте обобщенный класс Triangle, описывающий треугольник. Треугольник задан сторонами *a*, *b* и *c*. Класс должен содержать:

* поля,
* свойства с дополнительной логикой,
* конструктор,
* метод вывода информации о треугольнике,
* метод, возвращающий площадь треугольника.

Написать программу, иллюстрирующую работу данного класса. В качестве универсальных параметров используйте типы int, uint и double.

**Задание 44.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте обобщенный класс Segment, описывающий отрезок на координатной плоскости. Отрезок задан координатами точек (*x*1; *y*1), (*x*2; *y*2). Класс должен содержать:

* поля,
* свойства,
* конструктор,
* метод вывода информации об отрезке,
* метод, возвращающий длину отрезка.

Написать программу, иллюстрирующую работу данного класса. В качестве универсальных параметров используйте типы int, long и double.

**Задание 45.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте статический метод int Sum (int m, int n), который вычисляет и возвращает сумму всех целых чисел от *m* до *n* включительно.
2. Объявите тип делегата, соответствующий этому методу.
3. Объявите сам делегат.
4. Применить в программе созданный делегат 3 раза.

**Задание 46.** Используя Visual Studio, создайте проект по шаблону Console Application.

Определить тип Frequency – перечисление(enum) со значениями Weekly, Monthly, Yearly.

Определить класс Magazine (Журнал), который имеет открытые автореализуемые свойства, доступные для чтения и записи:

* свойство типа string c названием журнала;
* свойство типа Frequency с информацией о периодичности выхода журнала;
* свойство типа DateTime c датой выхода журнала;
* свойство типа int с тиражом журнала.

В классе Magazine определить конструкторы:

* конструктор с параметрами типа string, Frequency, DateTime, int для инициализации соответствующих свойств класса;
* конструктор без параметров, инициализирующий свойства класса значениями  
  по умолчанию.

Создать массив из *n* объектов типа Magazine, вывести полную информацию обо всех журналах.

Организовать поиск журналов по выбранному пользователем критерию. В случае отсутствия журналов, соответствующих критерию поиска, вывести об этом информацию.

**Задание 47.** Используя Visual Studio, создайте проект по шаблону Console Application.

Определить тип **Education** – перечисление(enum) со значениями Specialist, Вachelor, SecondEducation.

Определить класс **Student**  (Студент), который имеет открытые автореализуемые свойства, доступные для чтения и записи:

* свойство типа string c данными о Ф.И.О. студента;
* свойство типа Education с информацией о форме обучения;
* свойство типа DateTime – дата рождения студента;
* свойство типа int для номера группы.

В классе **Student**  определить конструкторы:

* конструктор с параметрами типа string, Education, DateTime, int для инициализации соответствующих свойств класса;
* конструктор без параметров, инициализирующий свойства класса значениями  
  по умолчанию.

Создать массив из *n* объектов типа **Student**, вывести полную информацию обо всех студентах.

Организовать поиск студентов по выбранному пользователем критерию. В случае отсутствия студентов, соответствующих критерию поиска, вывести об этом информацию.

**Задание 48.** Используя Visual Studio, создайте проект по шаблону Console Application.

Описать интерфейс ISolid для геометрических тел. Интерфейс должен содержать методы: Volume и SurfaceArea, возвращающие объём и площадь поверхности соответственно.

Описать классы Cube (куб) и Cylinder (цилиндр), реализующие этот интерфейс. Параметры тел должны задаваться при создании экземпляра.

Написать метод, принимающий тело и выводящий на экран её название, параметры, объём и площадь поверхности.

Написать программу, использующую этот метод.

**Задание 49.** Используя Visual Studio, создайте проект по шаблону Console Application.

Описать интерфейс IShape для геометрических фигур. Интерфейс должен содержать методы Perimeter и Area, возвращающие периметр и площадь соответственно.

Описать классы Triangle (треугольник) и Disk (круг), реализующие этот интерфейс. Параметры фигур должны задаваться при создании экземпляра.

Написать метод, принимающий фигуру и выводящий на экран её название, параметры, периметр и площадь.

Написать программу, использующую этот метод.

**Задание 50.** Используя Visual Studio, создайте проект по шаблону Console Application.

Описать интерфейс ICalculation для преобразований чисел типа double. Он должен содержать метод Perform (выполнить преобразование), принимающий число и возвращающий результат преобразования.

Описать два класса, реализующих этот интерфейс: Add и Multiply. Первый класс увеличивает число на некоторую величину, а второй умножает на заданный коэффициент (величины определяются конструкторами).

Написать метод Calculate, принимающий число и два преобразования. Метод должен последовательно применять преобразования к числу и возвращать результат.

Написать программу, использующую этот метод.

Пример вызова метода: double x = Calculate (1, new Add(2), new Multiply (3));

**Задание 51.** Используя Visual Studio, создайте проект по шаблону Console Application.

Класс **Zakaz** с полями **fam** (фамилия), **size** (количество мест в номере), **comfort** (комфортность), методом **Show** (показать заказ) и четырьмя конструкторами с разным числом и типами параметров. Модифицируйте программу, добавив возможность многократного ввода заказа с клавиатуры (фамилия, количество мест в номере, комфорт). Завершение ввода – символ **Q**. Создайте приложения, в которых определяются классы, поля, конструкторы, методы, создаются и инициализируются 2–3 объекта.

**Задание 52.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создайте класс Купюра.

2. Добавьте поля, например: название, номинал, количество.

3. Для каждого поля создайте свойство с методами доступа get и set.

4. Обеспечьте проверку правильности вводимых данных.

5. Добавьте метод, выводящий на экран значения всех полей класса и продемонстрируйте его работу.

6. Создайте три объекта данного класса. Задайте значения свойствам класса и выведите значения на экран.

7. Добавьте конструктор по умолчанию и конструктор, принимающий один или несколько параметров.

8. Создайте объекты данного класса с помощью различных конструкторов.

9. Добавьте в класс методы, например:

-   изменить количество купюр, увеличив или уменьшив его на указанное число;

-   определить, достаточна ли сумма для оплаты покупки указанной стоимости.

10. Обеспечьте возможность продемонстрировать правильность работы созданных методов.

**Задание 53.** Используя Visual Studio, создайте проект по шаблону Console Application.

Описать класс Point (*Точка*), описывающий точку на плоскости. Определить в нем:

1. конструктор с параметрами, принимающий координаты точки;
2. статические поля, определяющие:

– количество точек 1-й координатной четверти;

– количество точек 2-й координатной четверти;

– количество точек 3-й координатной четверти;

– количество точек 4-й координатной четверти;

– общее количество точек;

1. метод для вывода координат точки;
2. статический метод, выводящий количество точек, расположенных в каждой координатной четверти, и общее количество точек.

Написать программу, использующую этот класс и методы.

**Задание 54.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создать класс DateClass для работы с датой и разработать следующие элементы класса:

a. поле DateTime data;

b. конструктор без параметров;

с. конструктор, позволяющий установить заданную дату (конструктор с тремя параметрами – год, месяц, день);

d. свойство, позволяющее установить или получить значение поле класса (доступно для чтения и записи);

e. метод, позволяющий вычислить дату предыдущего дня.

2. Создать объект класса с использованием конструктора с параметрами, введенными с клавиатуры.

**Задание 55.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создать класс DateClass для работы с датой и разработать следующие элементы класса:

a. поле DateTime data;

b. конструктор без параметров;

с. конструктор, позволяющий установить заданную дату (конструктор с тремя параметрами – год, месяц, день);

d. свойство, позволяющее установить или получить значение поле класса (доступно для чтения и записи);

e. метод, позволяющий вычислить дату следующего дня.

2. Создать объект класса с использованием конструктора с параметрами, введенными с клавиатуры.

**Задание 56.** Используя Visual Studio, создайте проект по шаблону Console Application.

1. Создать класс DateClass для работы с датой и разработать следующие элементы класса:

a. поле DateTime data;

b. конструктор без параметров;

с. конструктор, позволяющий установить заданную дату (конструктор с тремя параметрами – год, месяц, день);

d. свойство, позволяющее установить или получить значение поле класса (доступно для чтения и записи);

e. метод, позволяющий определить сколько дней осталось до конца месяца.

2. Создать объект класса с использованием конструктора с параметрами, введенными с клавиатуры.

**Задание 57.** Используя Visual Studio, создайте проект по шаблону Console Application.

Ввести две даты и определить количество дней между ними.

**Задание 58.** Используя Visual Studio, создайте проект по шаблону Console Application.

Запросить с клавиатуры возраст человека и вывести на экран следующее сообщение: Ты родился в .... году

**Задание 59.** Используя Visual Studio, создайте проект по шаблону Console Application.

Определить возраст человека.

**Задание 60.** Используя Visual Studio, создайте проект по шаблону Console Application.

Создайте обобщенный список из названий цветов. Выполните следующие манипуляции со списком с выводом каждого изменения содержимого списка:

1) добавьте в список элементы строчными буквами;

2) выведите содержимое коллекции на экран;

3) преобразуйте все буквы элементов в прописные;

4) выведите содержимое коллекции на экран;

5) определите текущее количество элементов коллекции;

6) добавьте в коллекцию новый элемент в конец списка;

7) выведите содержимое коллекции на экран;

8) добавьте в коллекцию новый элемент в начале списка соответствующего варианта;

9) выведите содержимое коллекции на экран;

10) удалите третий элемент коллекции;

11) выведите содержимое коллекции на экран;

12) расположите элементы в обратном порядке