**ПРАКТИЧЕСКАЯ РАБОТА № 8.**

**Структура. Перегрузка методов и классов С#**

****

**Цель работы**: научиться разрабатывать СТРУКТУРЫ, их основныекомпоненты: поля, методы, конструкторы, создавать объекты класса, обращаться к открытым полям и методам СЬРУКТУРЫ.

Для защиты практической работы необходимо изучить теоретическую часть, выполнить практическое задание, ответить на вопросы.

|  |  |  |
| --- | --- | --- |
| №  п/п | Ф.И.О. | НОМЕР ВАРИАНТА/ЗАДАНИЯ |
|  | Абабий Илья | Набор примера из лекционного материала |
|  | Ботнарь Артем | 1 |
|  | Войнован Леонид | 2 |
|  | Гулин Максим | 3 |
|  | Дубчак Егор | 4 |
|  | Дяченко Ольга | Набор примера из лекционного материала |
|  | Игнатенко Максим | Набор примера из лекционного материала |
|  | Крижановская Арина | 5 |
|  | Лебедева Серафима | 6 |
|  | Лунгу Никита | 4 |
|  | Марков Кирилл | 7 |
|  | Мустяца Андрей | 8 |
|  | Мутаф Артем | 9 (дополнительные условия придумать самим) |
|  | Накул Вячеслав | 10 |
|  | Овчарук Дарья | 11 |
|  | Полозюк Даниил | 12 |
|  | Почтарь Александр | 13 |
|  | Ратушинский Михаил | 14 |
|  | Ривилис Евгений | 15 |
|  | Салтыков Антон | 5 |
|  | Сиглов Артем | 6 |
|  | Ткачук Анастасия | 14 (дополнительные условия придумать самим) |
|  | Цвигуненко Юлия | 11 (дополнительные условия придумать самим) |
|  | Черных Артем | 10 |
|  | Шурдук Виктория | 9 |
|  | Щуровская Александра | 8 |
|  | Янишевский Владимир | 7 |

**Теоретические сведения**

Наряду с методами мы можем также перегружать операторы. Например, пусть у нас есть следующий класс Counter:

Перегрузка операторов заключается в определении в классе, для объектов которого мы хотим определить оператор, специального метода:

Public static возвращаемый\_тип operator оператор(параметры)

{ }

Этот метод должен иметь модификаторы public static, так как перегружаемый оператор будет использоваться для всех объектов данного класса. Далее идет название возвращаемого типа. Возвращаемый тип представляет тот тип, объекты которого мы хотим получить. Затем вместо названия метода идет ключевое слово operator и собственно сам оператор. И далее в скобках перечисляются параметры. Бинарные операторы принимают два параметра, унарные - один параметр. И в любом случае один из параметров должен представлять тот тип - класс или структуру, в котором определяется оператор.

Поскольку все перегруженные операторы - бинарные - то есть проводятся над двумя объектами, то для каждой перегрузки предусмотрено по два параметра.

Стоит отметить, что так как по сути определение оператора представляет собой метод, то этот метод мы также можем перегрузить, то есть создать для него еще одну версию.

Следует учитывать, что при перегрузке не должны изменяться те объекты, которые передаются в оператор через параметры.

Поскольку оператор унарный, он принимает только один параметр - объект того класса, в котором данный оператор определен. Но это неправильное определение инкремента, так как оператор не должен менять значения своих параметров.

При перегрузке операторов надо учитывать, что не все операторы можно перегрузить. В частности, мы можем перегрузить следующие операторы:

унарные операторы +, -, !, ~, ++, --

бинарные операторы +, -, \*, /, %

операции сравнения ==, !=, <, >, <=, >=

логические операторы &&, ||

Как уже отмечалось ранее, в теле класса могут быть объявлены:

константы;

поля;

конструкторы и деструкторы;

методы;

события;

делегаты;

вложенные классы (структуры, интерфейсы, перечисления). Методы, называемые **свойствами (***Properties***)**, представляют

специальную синтаксическую конструкцию, предназначенную для обеспечения эффективной работы клиентов класса с его полями. При работе со свойствами объекта (полями) нужно решить, какую стратегию доступа использовать, чтобы обеспечить требуемый режим к полю класса со стороны клиентов класса. Наиболее часто используют следующие стратегии:

чтение, запись (*Read, Write*);

чтение, запись при первом обращении (*Read, Write-once*);

только чтение (*Read-only*);

только запись (*Write-only*);

ни чтения, ни записи (*NotRead, NotWrite*).

Чаще всего, поля класса объявляют закрытыми, а обращение к ним организуют через свойства.

При обращении к свойству со стороны клиента класса, набор

операторов, входящий в тело аксессора *set* будет выполняться при вводе данных, а набор операторов, входящий в тело аксессора *get* будет выполняться при чтении данных, при этом вводимые данные имеют имя *value*.

Свойства являются частным случаем метода класса, обладающим особым синтаксисом. Еще одним частным случаем является **индексатор**. Метод-индексатор является **обобщением метода-свойства. Он обеспечивает доступ к закрытому полю, представляющему массив**. Объекты класса индексируются по этому полю.

Синтаксически объявление индексатора - такое же, как и в случае свойств, но аксессоры *get* и *set* приобретают аргументы по числу размерности массива, задающего индексы элемента, значение которого читается или обновляется. Класс может иметь не более **одного индексатора,** имя индексатора известно заранее, это - *this*. В том случае, если класс использует несколько массивов, то индексация объектов может быть выполнена только по одному из них.

У класса могут быть поля, связанные не с объектами, а с самим классом. Эти поля объявляются как **статические** с модификатором *static*. **Статические поля** доступны всем методам класса. Независимо от того, какой объект класса обратился к статическому полю, организуется обращение к одному и тому же значению, определяемому статическим полем.

Вызов **статических полей** и **методов класса** его клиентами имеет свои особенности.

Так при обращении к обычному открытому полю/методу, необходимо создать объект класса, а затем, указав имя объекта, через точку указать имя поля/метода.

Для вызова открытого статического поля/метода необходимо указать имя класса и через точку имя поля/метода. Таким образом, обращение к открытым статическим полям/методам не требует создания объект данного класса.

Пример такого обращения, с использованием стандартной библиотеки классов приведен ниже.

Мы не создавали объект класса *Console*, а сразу обратились к его статическому методу *WriteLine*, во второй строке примера мы не создавали объект класса *Math*, а сразу обратились к его статическому полю *Pi*.

ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОГО ВЫПОЛНЕНИЯ

Разработать согласно заданию СТРУКТУРЫ. Предусмотреть конструктор по умолчанию и с параметрами. Создать методы, работающие с полями структуры. Предусмотреть наличие методов-свойств, статических полей и методов. Продемонстрировать работу объектов структуры.

Описания членов-данных пользовательских классов:

1. СТУДЕНТ: ФИО, курс, пол, оценки. Решить задачу определения студентов, которые будут получать стипендию, вычислить размер стипендии. Учесть, что стипендия начисляется в условных единицах (РУМЗП) и в момент расчета учитывается текущее значение РУМЗП. Величина стипендии

зависит от полученных оценок, но ряд студентов получают ее вне зависимости от оценок, например староста и студенты- сироты. В программе рассмотреть случай группы из 10 студентов.

* 1. СЛУЖАЩИЙ: имя, возраст, рабочий стаж, должности. Решить задачу начисления заработной платы (оклад, пропорционально количеству отработанных в этом месяце дней, премия1, премия2, вычеты, итого на руки).
* программе рассмотреть случай, для двух отделов, в которых работают по 5 человек. Процент премии 2 (от суммы начисленной з/пл) одинаков по всей организации.
  1. КАДРЫ: ФИО, номер цеха, разряд, специальность, должность. В зависимости от стажа работы на предприятии, каждому сотруднику выплачивается доплата (<5 лет – 0 руб., 5-10 лет *x*1 рублей, 10-15 лет – *x*2 рублей, больше 15 лет *x*3 рублей). Эти значения в разные месяцы могут меняться. Если объявляется рабочему выговор, то доплата не производится.
* программе рассмотреть случай 10 рабочих имеющих различный стаж работы. Сформировать ведомости на выдачу доплаты за 3 месяца.
  1. ИЗДЕЛИЕ: группа 1, группа 2, название, шифр, вес, комплектация, стоимость. Стоимость изделия задается в условных единицах и в момент продажи переводится в рубли по текущему курсу. Группа 1 может быть: малоценка, если стоимость меньше 10 у.е., средней цены, если стоимость от 10 до 1000 у.е., повышенной стоимости (от 1000 до 10000 у.е.), значительной стоимости (>10000 у.е.). Группа 2 может быть: легкое (до 500 грамм), средней тяжести (от 500 грамм до 5 кг), значительной тяжести (от 5 кг до 500 кг), очень тяжелые (> 500 кг). При продаже изделия, в сертификате указыва-ется название страны - получателя. Программа должна обеспечить формиро-вание в виде текста сертификата изделия с указанием всех его параметров и страны назначения. Рассмотреть случай, когда продают по 5-6 изделий в 3 страны.
  2. ПЕЧАТНОЕ ИЗДАНИЕ: название, ФИО автора, стоимость, оглавление.

Стоимость книги со временем меняется.

Для художественной литературы: первые 5 лет она равна стоимости покупки книги, следующие 5 лет уменьшается на 20% в год, пока не станет равной нулю, если возраст книги превышает 100 лет, то ее стоимость в k раз превышает первоначальную стоимость.

Для прочих книг: первые 5 лет она равна стоимости покупки книги, следующие 5 лет уменьшается на 20% в год, пока не станет равной нулю.

Решить задачу по учету 10 книг, выдаваемых различным читателям, возможна в случае порчи замена книги на книгу с не меньшей стоимостью.

Программа должна: показать не выданные книги; Показать все книги данного автора; Дать информацию о замене книг.

6. ЭКЗАМЕН: ФИО студента, дата, оценка, перечень вопросов.

Создать класс «Билет», в котором указать 2 вопроса и задачу, а также зафиксировать ФИО студента и результаты его ответов. Организовать прове-дение экзамена:

- выбор студентам одного из 10 билетов, за каждый вопрос студент по-лучает некую оценку;

- если оценка за один пункт билета равна «неуд», то задается дополни-тельный вопрос (преподаватель имеет 3 любимых вопроса);

* если оценки за два пункта билета «неуд», студент получает общую оценку «неуд» и ему назначается дата пересдачи.

Программа должна выдать на экран результаты экзамена (5 студентов) в кратком виде (только оценки) и в полном виде (оценки за каждый вопрос билета + дополнительны = общая оценка).

7. АДРЕС: город, улица, номер дома, номер квартиры, № телефона, список жильцов.

Написать программу, которая использует объекты класса «адрес». Программа формирует адреса 10 человек и по запросу пользователя выдает на экран:

* по заданной фамилии, имени, отчеству - адрес;
* по заданному адресу (город, улица, дом) – ФИО;
* по номеру телефона – адрес и ФИО;

Учесть то, что названия улиц могут быть изменены.

1. ТОВАР: название, артикул, стоимость, даты (изготовление, срок реализации).

Написать программу, использующую объекты класса «товар». Учесть, что стоимость товара задается в условных единицах и только в момент продажи умножается на соответствующее значение. Кроме того, покупателю может быть предоставлена скидка (0-50%). Программа должна предоставлять пользователю информацию о 10 товарах, пользователь оформляет покупку выбранного товара. Если сумма покупки превышает некоторое значение, автоматически формируется некая скидка. Продавец по своему желанию может указать свое значение скидки.

Программа должна формировать список закупленных товаров с указанием покупателя, количества и вида товара, суммы и предоставленной скидки.

1. ЦЕХ: название, начальник, кол-во рабочих, перечень номенклатуры выпускаемых изделий.

В зависимости от разряда рабочего меняется его оплата (1-й разряд – 100% стоимости выполненной работы, 2-й 110%, 3-й 120, 4-й 130%, 5-й 150%). Эти значения в разные месяцы могут меняться. Если объявляется рабочему выговор, то доплата не производится. Рабочий, не имеющий выговора, может получить премию, которая задается для всех одинаково в рублях. В программе рассмотреть случай 10 рабочих имеющих различный стаж работы. Каждый рабочий может изготавливать 5 различных изделий. За каждое изделие он получает некоторую сумму. Рассчитать зарплату 10 рабочим.

1. АВТОМОБИЛЬ: марка, цвет, мощность двигателя, расход горючего на 100 км, стоимость, виды и даты ремонтов.

Написать программу, позволяющую вести набор данных о состоянии 10 автомобилей. Дается некий список видов ремонтов (не менее 5) с указанием стоимости ремонта и его длительности. Программа должна фиксировать ве-личину пробега каждого автомобиля (каждый день), проведенные ремонты. Рассчитать стоимость эксплуатации каждого автомобиля в течение 10 дней. Учесть, что стоимость горючего для всех автомобилей одинакова.

1. ОБЛАСТЬ: страна, название области, областной центр, форма прав-ления, площадь, население, список районов, основные отрасли экономики. Написать программу, которая поддерживает справочник областей страны и позволяет произвести несколько (не менее 3-х) видов поиска. Учесть, что данные меняются от года в год. Справочник содержит информацию о странах за 10 лет.
2. КОРАБЛЬ: название, год и место постройки, водоизмещение, тип, вид корабля, порт приписки, грузоподъемность, количество мест для пасса-жиров (1, 2, 3 го классов), расход топлива в день при: экономичном ходе, максимально ходе; запас топлива.

Создать класс «корабль».

Написать программу- справочник кораблей, содержащую данные о 10 кораблях. Произвести расчет стоимости перехода на указанное расстояние, для указанной даты, для всех кораблей. Учесть, что стоимость топлива для всех кораблей одинакова.

13.СТУДЕНТ: ФИО, курс, пол, группа, специальность. Решить задачу определения числа студентов, которые: а) обучаются в заданной оператором группе; б) изучают заданную оператором специальность. Сформировать массив объектов класса Студент. Использовать статические поля. Статический метод должен выдавать название вуза.

14.СПИСОК ЖИЛЬЦОВ: ФИО, № квартиры, улица, № дома. Решить задачу нахождения числа жителей: а) заданной квартиры; б) заданного дома; в) города. Использовать статические поля. Статический метод должен выдавать название страны.

15.АВТОМОБИЛЬ: марка, цвет, мощность двигателя, расход горючего на

100 км, год изготовления, стоимость.

Написать программу, позволяющую вести набор данных о 10 автомоби-лях. Рассчитать стоимость автомобиля. Учесть, что эта стоимость состоит из стоимости, заданной владельцем автомобиля и налога, который зависит от даты изготовления автомобиля. Например, если возраст автомобиля >10 лет, то это 20% его стоимости, если <=10 лет, то это 10% его стоимости. Статиче-ский метод должен выдавать название страны, в которой производится рас-таможивание автомобиля.

КОНТРОЛЬНЫЕ ВОПРОСЫ:

1. Методы-свойства класса. Назначение и описание.
2. Стратегии, реализуемые свойствами.
3. Роль аксессоров*set* и *get*.
4. Назначение, описание и вызов статических методов.
5. Описание и вызов статических переменных.
6. Индексаторы. Назначение и описание.
7. В тексте программы присутствуют следующие строки:

*int k;*

*k=classA.M1(100);*

*Itrw h=new Itrw();*

*h.tt=k;*

*if (h.tt!=k)*

*Console.WrineLine(“Error!”);*

Прокомментируйте их.