

В.В. Новый

Современные технологии разработки программного обеспечения

Практикум по дисциплине «Современные технологии разработки программного обеспечения» для студентов 3 курса специальности Прикладная математика (1-31 03 03)

2010

Лабораторная работа №1
Создание приложения для .NET Framework. Базовые типы данных.
(2 часа)

Инструменты разработчика на платформе .NET

Для начала разработки программ на C# достаточно наличия установленной платформы Microsoft .NET Framework (dotnetfx.exe) и любого текстового редактора. Платформа Microsoft .NET Framework содержит все необходимое для компиляции, отладки и запуска программы на C#.

После установки платформы Microsoft .NET Framework ее файлы можно найти в подкаталоге Microsoft.NET\Framework\ каталога Windows. При этом может быть установлено несколько версий платформы одновременно в различных подкаталогах названного каталога. Например, в подкаталоге v4.0.30319 – версия платформы .NET 4.0.

Основными программными инструментами разработчика и имеющими к ним отношение файлами, «обитающими» в этом каталоге являются:

- csc.exe – компилятор командной строки для языка C#;
- vbc.exe – компилятор Visual Basic.NET;
- jsc.exe – компилятор JScript.NET;
- ilasm.exe – ассемблер для IL;
- MSBuild.exe – утилита сборки проекта от Microsoft;
- ngen.exe – утилита создания «нативного» кода из сборки .NET;
- csc.rsp – файл ответов для компилятора csc.exe, содержит ключи командной строки для подключения основных сборок компилируемой программе на C#.

Другой, важный для платформы .NET каталог, - каталог %Windir%\Asssembly. Здесь располагается GAC (Global Assembly Cache – глобальный кэш сборок) – каталог размещения общих сборок.

К другим полезным инструментам, расположенным в других каталогах относятся, например:

- ildasm.exe – Intermediate Language Disassembler utility – утилита дизассемблера IL, располагающаяся в подкаталоге bin каталога установки SDK (минимальная версия SDK устанавливается с Microsoft Visual Studio Express Edition). Она позволяет загрузить любую сборку .NET и просмотреть ее манифест, программный код и метаданные;
- gacutil.exe – утилита управления GAC; расположена в том же каталоге, что и ildasm.exe.

Кроме поставляемых Microsoft средств разработки, существует много свободных программных проектов, посвященных .NET-разработке:

NUnit – утилита тестирования .NET-модулей, являющаяся аналогом JUnit для Java;

NAnt – конкурирующая с MSBuild утилита компоновки .NET, аналогичная Ant, используемой в Java;

NDoc – утилита генерации документации на базе XML-комментариев в .NET, аналогичная по назначению JavaDoc и позволяющая получать документацию в форматах XML, HTML, LaTeX, chm и др.;

и т.д.

Задание 1. Определите, какие версии платформы Microsoft .NET Framework установлены на Вашем компьютере. Найдите компилятор языка C#.

Создание и компиляция программы на языке C#

После написания программы на языке C# сохраните полученный файл с расширением “cs”. Следующим Вашим шагом является компиляция исходного кода программы в компоновочный блок среды CLR – сборку. Для этого Вам потребуется консольный компилятор языка C# - csc.exe:

«путь к файлу\csc.exe» «путь к файлу исходного кода\файл_исходного_кода.cs»

Для удобства набора, Вы можете выбрать один из следующих вариантов:

- 1) добавить путь к каталогу, содержащему компилятор в переменную окружения PATH;
- 2) запускать компиляцию через командную строку Microsoft Visual Studio 2008 (запустить ее через Программы\Microsoft Visual C++ 2008 Express Edition\Visual Studio Tools\Visual Studio 2008 Command Prompt)
- 3) Написать cmd-файл, содержащий нужную командную строку с указанными путями. Например:

```
C:\WINDOWS\Microsoft.NET\Framework\v4.0.30319\csc.exe /target:exe S:\MyCSharpProjects\Demo\SupaDupaClass.cs
```

И компилировать программу запуском этого cmd-файла.

Ключи запуска компилятора csc.exe можно посмотреть, набрав в командной строке «csc.exe /? «.

Наиболее интересные из них это:

`/out` – используется для задания имени создаваемой сборки. По умолчанию имя компоновочного блока совпадает с именем файла исходного кода *.cs или с именем файла, содержащего метод Main(), если компилируется несколько файлов.

`/target:тип_сборки` – указывает вид создаваемого компоновочного блока. Здесь тип_сборки может принимать следующие значения:

`exe` – консольное приложение. Значение по умолчанию, поэтому может не указываться явно;

`library` – создание библиотеки кода *.dll;

`winexe` – создание Windows-приложения (также может создаваться с целевым значением exe, но при этом будет создано и консольное окно);

`module` – создание компоновочного модуля для многомодульныхборок.

Задание 2. Наберите пример «Hello, World!» в любом текстовом редакторе (рекомендуется Notepad++). Откомпилируйте полученный исходный код с ключами компилятора /target:exe и /target:library.

Задание 3. С помощью утилиты ildasm.exe изучите полученные компоновочные модули: исходный код на IL, метаданные сборок и их манифест.

Документирование исходного кода

В отличие от javadoc в Java, в языке C# модель документации несколько отличается. Преобразованием комментариев исходного кода в XML занимается не отдельная утилита, а компилятор csc.exe (опция командной строки /doc).

Для задания комментариев используются т.н. документирующие комментарии – тройной слэш (///). Внутри документирующих комментариев допускается использование элементов XML-комментариев кода:

- <code> - множество строк должны быть помечены как код;
 - <list> - вставка списка или таблицы в файл документации;
 - <param> - описание заданного параметра;
 - <returns> - описание возвращаемого значения;
 - <see> - перекрестная ссылка на связанные элементы в документации;
 - <summary> - итоговая информация для заданного элемента;
- и т.д.

При использовании для разработки IDE Visual Studio (Express Edition) среда генерирует «скелет» документации кода при указании документирующего комментария.

Рассмотрим в качестве примера комментарии к методу, суммирующему два числа:

```
/// <summary>
/// Метод для суммирования двух целых чисел
/// </summary>
/// <param name="first">первое слагаемое</param>
/// <param name="second">второе слагаемое</param>
public int Sum(int first, int second)
{
    return first + second;
}
```

Чтобы сгенерировать документацию, используют следующую командную строку подобную заданной:

```
csc /doc:ИмяXMLФайла.xml ИмяФайлаПрограммы.cs
```

К сожалению, в Visual Studio не предусмотрена утилита для преобразования xml-файла документации в более дружелюбный к пользователю HTML автоматически. Вместо этого можно использовать свободную утилиту NDos, позволяющую преобразовать документацию в нужный формат.

Задание 4. Добавьте подробные комментарии к проекту «Hello, World!» и сгенерируйте для него XML-документацию.

Задание 5. Разработайте программу, вычисляющую и выводящую на экран в виде таблицы значения функции, заданной с помощью ряда Тейлора, на интервале от X_1 до X_2 с шагом dx и точностью h . Каждая строка таблицы должна содержать значение аргумента, значение функции и количество просуммированных членов ряда.

Вариант 1.
$$e^{-x} = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^n}{n!}, |x| < \infty$$

Вариант 2.
$$\ln(1-x) = -\sum_{n=1}^{\infty} \frac{x^n}{n}, -1 \leq x < 1$$

Вариант 3.
$$\cos(x) = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n}}{(2n)!}, |x| < \infty$$

Вариант 4.
$$\frac{\sin(x)}{x} = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n}}{(2n+1)!}, |x| < \infty$$

Вариант 5.
$$e^x = \sum_{n=0}^{\infty} \frac{x^n}{n!}, |x| < \infty$$

Вариант 6.
$$\ln(x) = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot (x-1)^{n+1}}{(n+1)}, 0 < x \leq 2$$

Вариант 7.
$$\ln(x+1) = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{n+1}}{n+1}, -1 < x \leq 1$$

Вариант 8.
$$\operatorname{arctg}(x) = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n+1}}{(2n+1)}, |x| \leq 1$$

Вариант 9.
$$\sin(x) = \sum_{n=0}^{\infty} \frac{(-1)^n \cdot x^{2n+1}}{(2n+1)!}, |x| < \infty$$

Вариант

10.

$$\operatorname{ch}(x) = \sum_{n=0}^{\infty} \frac{x^{2n}}{(2n)!}$$

Лабораторная работа №2
ООП в C#
(4 часа)

Задание 1. Разработайте информационную систему для решения задачи Вашего варианта.

Вариант 1. Учет багажа авиарейса.

Система учета багажа авиарейса содержит информацию о двух типах багажа: багаж пассажиров и дипломатический багаж. Каждая запись системы содержит: номер рейса, дата и время вылета, пункт назначения, количество мест багажа, суммарный вес багажа. Кроме того, в зависимости от типа багажа содержится информация о ФИО пассажира или стране-владельце дипломатического багажа. Реализовать регистрацию и удаление багажа в системе, а также поиск по номеру рейса, дате вылета, пункту назначения, весу багажа (превышение максимально допустимого).

Вариант 2. Учет платежей пункта коллективного пользования.

Система учета платежей пункта коллективного пользования поддерживает информацию о 2 видах платежей: оплата телефонного звонка и оплата доступа в Интернет. Каждая запись содержит: код записи, дату, тариф за минуту, скидка (в процентах), время начала, время окончания. При этом для оплаты телефонного звонка поддерживается информация о вызываемом номере, а для услуги доступа в Интернет ФИО и адрес клиента. Реализовать добавление записей в систему и подготовку чека об оплате, а также поиск по дате, времени пользования услугой, длительности сеанса.

Вариант 3. Товары на складе.

Система управления складом хранит информацию о продовольственных и непродовольственных товарах. Общая информация включает в себя: номер склада, код товара, наименование товара, дата поступления на склад, срок хранения, количество единиц товара, цена за единицу товара. Кроме того, информация о продовольственных товарах содержит данные об условиях хранения товара. Реализовать добавление и удаление товаров со склада, а также поиск по номеру склада, коду товара, дате поступления и сроку хранения/реализации (просроченные и непросроченные товары).

Вариант 4. Учет транспорта на автостоянке.

Журнал учета транспорта на автостоянке состоит из карточек, описывающих технику: легковые автомобили, грузовики, спец. автотранспорт. Карточка содержит как общую информацию (номер транспортного средства, ФИО владельца, дата и время начала и окончания, тариф за час), так и специфическую (например, тип кузова для легкового автомобиля, тип шасси для спецавтотранспорта). Требуется реализовать регистрацию и удаление транспорта на автостоянке с подсчетом суммы к оплате, а также поиск по

номеру транспортного средства, по дате и времени стоянки, по фамилии владельца.

Вариант 5. Библиотека.

Каталог библиотеки хранит информацию о двух типах издания книг и периодике в фонде библиотеки. Каждая запись содержит следующую информацию: код, год выхода издания, наименование, количество страниц, наличие иллюстрации, издательство. Кроме того, записи о книгах содержат информацию об авторах книги, а записи о периодике – номер издания в этом году и общий номер. Требуется реализовать добавление и удаление издания в фонд библиотеки, а также поиск по коду, году, издательству и в зависимости от вида – номеру или автору.

Вариант 6. Платежи за коммунальные услуги.

Система учета платежей за коммунальные услуги поддерживает данные о счетах по квартплате, оплате газа, воды и электричества. Каждый счет содержит номер дома, номер квартиры, ФИО владельца, дату платежа, сумму платежа, процент пени, на сколько просрочен платеж. Кроме того, поддерживается информация, специфическая для типа платежа (размер квартиры в метрах квадратных, расход воды в литрах с учетом подогрева и без, объем газа в кубометрах и данные о состоянии счетчика электроэнергии). Реализовать добавление счетов, подсчет общей суммы за месяц, а также поиск по номеру дома, квартиры, владельцу, виду платежа, по дате.

Вариант 7. Счет в банке.

Банковская система поддерживает счета двух видов: счета физических лиц и счета юридических лиц. Счета физических лиц содержат: номер счета, код счета, ФИО владельца счета, сумму на счете, дату открытия счета, годовой процент начисления. Счета юридических лиц содержат: номер счета, код счета, название юридического лица, место регистрации юридического лица, сумму на счете, дату открытия счета, годовой процент начисления. Требуется реализовать процедуру открытия и закрытия счета, а также поиск по номеру счета, коду счета и владельцу или названию юридического лица.

Вариант 8. Информационная система по комплектующим компьютера.

Система хранит информацию о комплектующих в виде: код, торговая марка производителя, цена, количество экземпляров. Кроме того, в зависимости от типа комплектующих поддерживается дополнительная информация: тип и тактовая частота процессора, объем модулей памяти, объем и тип жесткого диска, объем памяти видеокарты и т.д. Реализуйте добавление и удаление комплектующих, а также поиск в базе по специфическим характеристикам комплектующих. Предусмотрите возможность автоматически составить: самую дорогую конфигурацию, самую дешевую конфигурацию,

конфигурацию для корпоративных заказчиков (по максимальному количеству одинаковых комплектующих).

Вариант 9. Учет договоров кабельного телевидения.

Система учета договоров кабельного телевидения поддерживает два типа договоров: договор по подключению телевизионных каналов и договор по предоставлению доступа в Интернет. Каждый договор содержит: ФИО, район, адрес, телефон клиента, номер договора, дата заключения договора, оплата за установку, абонентская оплата ежемесячно, дата последнего платежа. Кроме того, в зависимости от типа договора содержится дополнительная информация: список подключенных каналов, тарифный план и объем трафика. Требуется реализовать заключение и расторжение договоров, подготовку чека с суммой к оплате, а также поиск по району, дате заключения договора, дате последнего платежа.

Вариант 10. Туристическая фирма.

Туристическая фирма продает круизы на лайнерах по океану и путевки на курорты. Каждая запись содержит информацию о наименовании тура, ФИО клиента, цену одного дня, количество дней, курс валюты. Кроме того для круиза поддерживается информация о типе каюты, названии лайнера, а для путевки на курорт – название курорта, «звездность» гостиницы и стоимость проезда. Реализуйте добавление в систему новых записей, подсчет стоимости продаваемой услуги, а также поиск по наименованию тура, по стоимости, по количеству дней и для путевок на курорты – по стоимости проезда.

Лабораторная работа №3
Библиотека FCL
(4 часа)

Задание 1. Для проекта из предыдущей работы реализуйте хранение данных с помощью одной из коллекции System.Collections (или обобщенной коллекции), а также реализуйте сохранение и загрузку в файл с использованием средств ввода-вывода .NET.

Репозиторий ВГУ

Лабораторная работа №4
Делегаты. События. Отражение
(2 часа)

Задание 1. Дополните предыдущую лабораторную работу возможностью выбора способа сохранения данных на диск: в бинарном формате или в текстовом формате. Выбор нужного способа сохранения данных на диск реализуйте с помощью делегата.

Репозиторий ВГУ

Лабораторная работа №5
GUI-приложения на базе Windows Forms
(4 часа)

Мини-разминка: Разработайте приложение – простейшую головоломку, представляющее собой матрицу 4x4 из элементов (кнопок, меток или др.), которые способны принимать 2 значения (например, 0 и 1 или «+» и «-»). Первоначально все кнопки заполните случайными значениями. При нажатии на любую из кнопок все значения в ряду и столбце этой кнопки переключаются на противоположные. Пользователю требуется привести все значения к одному и тому же виду.

Задание 1. Для проекта из предыдущей лабораторной работы разработайте графический интерфейс пользователя на основе Windows Forms (проект Windows Forms Application). Предусмотрите выбор файлов для загрузки и сохранения с помощью диалогов.