

Співаковський О.В., Круглик В.С.
Херсонський державний університет

Технології розробки програмних засобів, які підтримують компонентно-орієнтований підхід

Світовий процес переходу від індустріального до інформаційного суспільства, а також соціально-економічні зміни, що відбуваються в Україні, вимагають суттєвих змін у багатьох сферах діяльності людей. В першу чергу це стосується системи освіти. Національною програмою "Освіта. Україна XXI" сторіччя передбачено забезпечення розвитку освіти на основі нових прогресивних концепцій, запровадження у навчально-виховний процес новітніх педагогічних технологій та науково-методичних досягнень, створення нової системи інформаційного забезпечення освіти, входження України у трансконтинентальну систему комп'ютерного інформування.

Для досягнення зазначених результатів необхідно швидкими темпами розвивати дистанційну освіту, запровадження якої в Україні передбачено Національною програмою інформатизації.

Особливо актуальним є застосування сучасних інформаційних технологій у тих сферах розумової діяльності, які є найбільш складні для сприйняття, коли складність навчання обумовлюється великою кількістю рутинної роботи. Велика кількість обчислень, яка супроводжує відшукування розв'язування тієї чи іншої задачі, не дає можливості учневі засвоїти сутність досліджуваних процесів і явищ, і як наслідок – не формує необхідних знань та вмінь. У національній державній програмі "Освіта" (Україна XXI століття) зазначено, що освіта має забезпечувати всебічний розвиток людини як цілісної особистості, її здібностей та обдарувань, збагачення на цій основі інтелектуального потенціалу народу, його духовності і культури, формування громадянина України, здатного до свідомого суспільного вибору.

Розвиток нових інформаційних технологій досяг того рівня, що можливим стала підтримка пізнавальної діяльності і поширення відповідних технічних засобів (комп'ютерних лабораторій, засобів телекомунікацій, оперативної поліграфії, систем інтерактивного відео, баз даних і т.д.), в навчальних закладах почався наступний етап цього процесу. Його основний зміст:

- активне засвоєння і фрагментарне втілення засобів нових інформаційних технологій в традиційні навчальні дисципліни і на цій основі – масове засвоєння викладачами нових методів і організаційних форм навчальної діяльності;
- постановка питання про радикальний перегляд змісту освіти, традиційних форм і методів навчально-виховної роботи;
- розробка і засвоєння систем навчально-методичного забезпечення ("програмно-методичних комплексів", "комп'ютерних кур-

сів”), які містять програмні засоби для ЕОМ, різноманітні відео і аудіо матеріали, тексти для учнів і методичні матеріали для викладачів.

Багато які традиційні комп’ютерні курси математики, фізики та інших дисциплін базуються на ідеях програмованого навчання, хоча й використовують усі апаратні та програмні засоби сучасної обчислювальної техніки та нові методи подання навчального матеріалу. Найбільш розвинутою та досконалою як з методичної, так і з технічної точки зору при цьому виявляється лекційна частина курсу. Як правило, учбовий матеріал лекційної частини дисципліни супроводжується системами контрольних питань і тестових завдань. Так за допомогою комп’ютерної системи забезпечується контроль знань.

Але, як відомо, навчальний план дисципліни включає ще й практичну частину – цикли практичних занять і лабораторних робіт. Формування практичних вмінь та навичок досягається саме тут, і ця частина навчального плану в багатьох дисциплінах (математика, фізика, інформатика, ...) є центральною. Іншими словами, студент знає математику, якщо він вміє розв’язувати математичні задачі. Проблема адекватної комп’ютерної підтримки практичних занять менше розроблена і є досить актуальною [2].

Для дистанційного навчання важливого значення набувають педагогічні програмні засоби. Для репродуктивного засвоєння навчального матеріалу, коли застосовуються друковані, аудіо- та відео- матеріали можливо обійтися і без педагогічних програмних засобів. В даному випадку дистанційне навчання в принципі виконує свою функцію при вивченні таких дисциплін, як історія, література, право, і т.п.

Але цього недостатньо при вивченні таких предметів, як математика, хімія, фізика, де важливу роль відіграє розв’язування задач. При вивченні таких дисциплін особливої ваги набуває практична діяльність учнів.

В мережі Інтернет педагогічні програмні засоби з підтримкою практичної діяльності учнів зустрічаються дуже рідко, тому актуальною є проблема створення технології їх розробки та впровадження для дистанційного навчання.

Сучасні педагогічно-орієнтовані інформаційні технології повинні забезпечувати комп’ютерну підтримку подання теоретичного матеріалу, формування практичних вмінь та навичок та контролю засвоєння теоретичних знань, практичних вмінь та навичок.

Технологія створення інтегрованих педагогічних програмних засобів для дистанційного навчання з підтримкою практичної діяльності учнів повинна задовольняти наступні вимоги:

- Предметна незалежність;
- Кроссплатформенність;
- Легкість впровадження існуючих програмних засобів.

Для реалізації даного підходу було створено програмну оболонку «Світ лінійної алгебри». Розглянемо детальніше підходи, що застосовувалися при розробці.

Компонентний підхід до побудови систем дистанційної освіти

Для зменшення собівартості підготовки систем для дистанційної освіти необхідно виділити загальні функції для всіх ППЗ та написати специфічні компоненти для кожної з них. При цьому важливо компоненти інтегрувати в оболонку, яка б забезпечувала підтримку наступні функції: легка інтеграція існуючих модулів, легка модернізація існуючих модулів, можливість розширення системи за рахунок нових модулів, контроль прав доступу до модулів (читання та зміна).

Аналіз конкретної предметної галузі дозволяє виділити компоненти, які необхідні для правильної та повноцінної організації учбової діяльності. Надалі розглянемо більш детально окремі компоненти та їх реалізацію.

Компонент для реалізації входу до системи. За допомогою цього компоненту реалізується аутентифікація користувачів та розподіл користувачів на групи.

Компонент безпеки. Використовується для авторизації користувачів та розподіл прав доступу в залежності від групи, до якої належить користувач.

Компонент адміністрування. Використовується для загального управління всім вузлом дистанційної освіти та організації взаємодії окремих компонент. Функціональність також дозволяє додавання, видалення, редагування даних про користувачів, визначення прав користувачів на доступ до окремих компонентів.

Ці три модулі грають особливу роль, оскільки утворюють потужний каркас, який забезпечує легке впровадження та зручне поєднання компонент, призначених для організації процесу навчання. Розглянемо ці компоненти.

Компонент „Навчальний посібник”. Являє собою модуль для роботи з структурованим гіпертекстом. За допомогою нього студент отримує необхідну теоретичну допомогу.

Компонент „Задачник”. Є сховищем задач для конкретного предмету. Може бути структурно та функціонально об'єднаним з компонентом „Учбовий посібник”. Забезпечено можливості викладачам додавати та редагувати завдання, а студентам копіювати завдання до свого зошиту.

Компонент „Зошит”. Служить для зберігання задач студента та роботи з ними. Дозволяє отримати відомості про те, чи розв'язана задача, чи самостійно розв'язана чи ні, оцінку за розв'язування. З „Зошита” студент також може завантажити задачу в середовище для розв'язування задач.

Компонент „Середовище для розв'язування задач”. Являє собою вузькоспеціалізований програмний модуль для розв'язування деякого класу задач конкретного предмету. Важливою особливістю

середовища розв'язування є, що воно повинна містити інструментарій для розв'язування задач даної конкретної галузі та вміти перевіряти розв'язок на кожному кроці розв'язування, забезпечуючи допомогу в разі необхідності. Зовнішній інтерфейс взаємодії з зошитом від зміни предмету змінюватися не повинен.

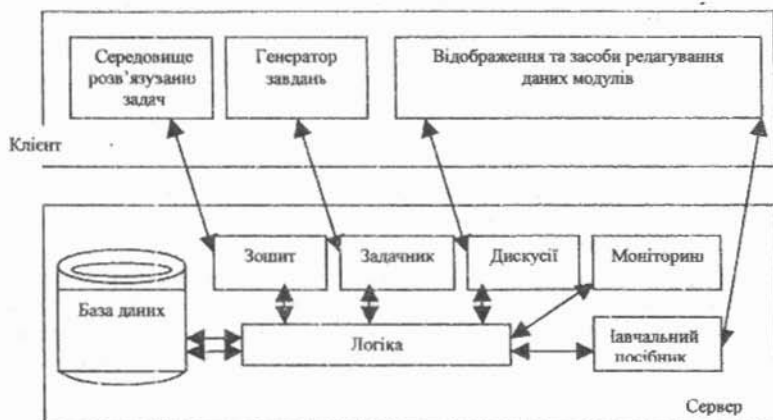
Компонент для проведення моніторингу. Дозволяє збирати дані та проводити статистичний аналіз для постійного контролю за ефективністю навчання.

Компонент для групового обговорення проблем. Організовується у вигляді форуму, де можна задати питання та отримати відповідь на нього.

Генератор завдань. Компонент призначено для генерації завдань в задачник. Працювати з компонентом може лише викладач.

Для успішної реалізації наведених вище компонентів і розміщення їх в мережі Інтернет виберемо за основу архітектуру клієнт-сервер. В даному випадку програмою клієнтом є Web-браузер. Постає питання, які з компонентів потрібно розмістити на сервері, а які на клієнті, а також де саме розмістити дані, з якими працює користувач. Ці питання були вирішені наступним чином. Всі дані, з якими працює користувач, розміщені і зберігаються на сервері. Клієнт лише має засоби для контролю над даними, інструментарій для управління даними. До клієнтської частини було віднесено такі компоненти: середовище для розв'язування задач та генератор завдань. Такий вибір викликаний вимогою до швидкодії. Якщо ці компоненти розміщені на клієнті, то всі операції над даними (формування та розв'язування задач) відбуваються без передавання даних на сервер, що дуже важливо для підвищення швидкодії при повільному Інтернеті.

Схема взаємодії клієнт-сервер, що використовується



Вибір технологій для розробки курсу дистанційної освіти

Дистанційні курси для мережі Інтернет можуть розроблятися з використанням веб-браузера в якості клієнта. Важливим є питання про вибір найкращої існуючої технології, яка б забезпечувала необхідну "інтерактивність" та можливості об'єктно-орієнтованого програмування, можливість повторного використання коду. Крім того, розробка та подальша підтримка не повинна бути надто складною.

Для розробки курсу було обрано такі інструментальні засоби: Microsoft Visual Studio 2003, Delphi 6.0; графічні редактори: CorelDraw 11, Adobe PhotoShop 7.0; підготовка текстів – Microsoft Word XP.

Операційна система – Microsoft Windows XP Professional.

Web-сервер – Microsoft Internet Information Server.

Серед обраних технологій можна назвати ASP.NET, XML, XSL, ADO.NET, COM+, ActiveX.

Розглянемо детальніше технологію ASP.NET, її переваги над іншими існуючими технологіями програмування для Інтернет.

По-перше, ASP.NET є системою, яка повністю підтримує концепції об'єктно-орієнтованого програмування.

По-друге, ASP.NET є системою, що підтримує модель програмування, засновану на подіях.

По-третє, в ASP.NET є вбудована підтримка XML формату даних.

ADO.NET в свою чергу забезпечує роботу з базами даних.

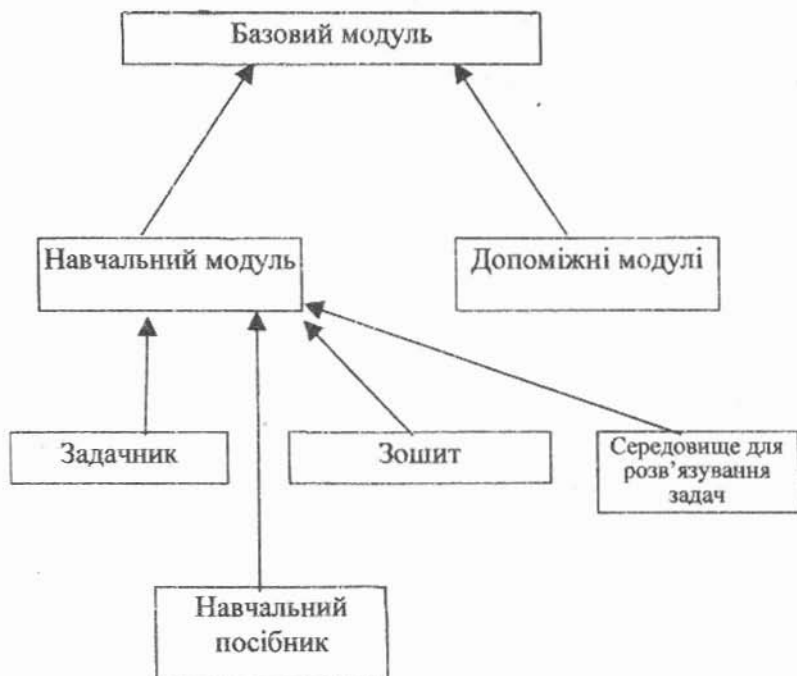
В якості мови програмування було обрано C#.

Подання даних в форматі XML забезпечує необхідну універсальність та кроссплатформенність при зберіганні та опрацюванні даних. XML є текстовим форматом описання даних. Він приніс нові методи передавання даних в Web додатках, забезпечуючи передавання та використання даних практично будь-яких типів де завгодно. Вся цінність XML витікає всього з двох простих особливостей. Перша особливість – XML можна розширювати. Це значить, що його можна легко доповнити, додавши власні структури та дескриптори користувача. По-друге, його формат є текстовим, що дозволяє створювати та редагувати його в будь-якому текстовому редакторі, наприклад в Блокноті.

Використання перетворень XSL дає можливість різного подання даних на смак користувача, використовуючи при цьому як файл даних XSL.

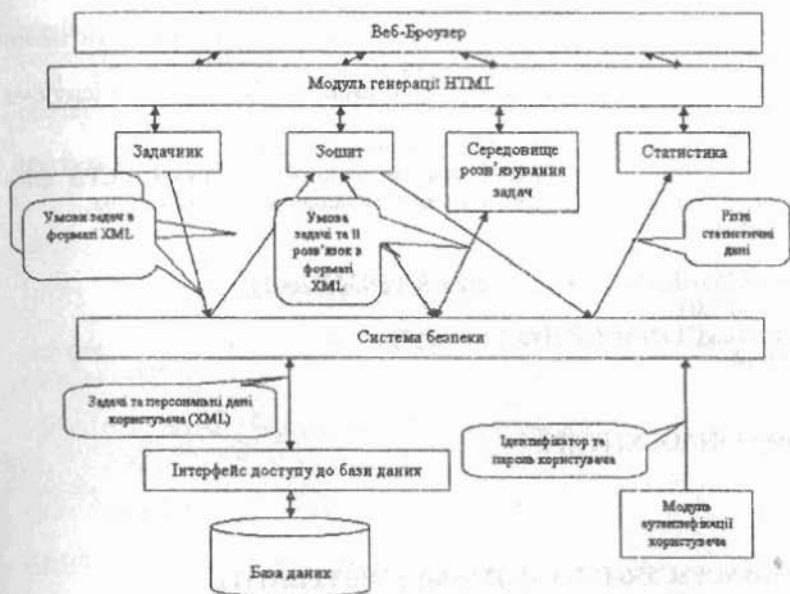
Основою будь-якої корпоративної системи є сховища даних. Тому правильний вибір типу сховища гарантує високу швидкість та стабільність роботи системи. В нашому випадку як корпоративне сховище даних було обрано Microsoft SQL Server 2000. Ця СУБД відповідає усім вимогам щодо швидкості, безпеки, легкості адміністрування. Використання збережених процедур сприяє швидкості виконання запитів та зменшенню кількості помилок при роботі з системою взагалі.

Ієрархія модулів для використання в системах дистанційного навчання



Модулі реалізовані у вигляді користувацьких компонентів системи ASP.NET.

Користувацькі компоненти системи ASP.NET – це сукупність елементів HTML, які об'єднані загальною функціональністю. Вони служать для візуального подання повідомлень користувачам. Формування даних проходить в компонентах логіки, які можуть безпосередньо зв'язуватися з базою даних. Таким чином можна побудувати наступний ланцюг передавання даних:



Компонент „Середовище для розв'язування задач”

„Середовище для розв'язування задач” (надалі СРЗ) є уніфікованим середовищем, розробленим під певну предметну галузь, яке надає необхідний інструментарій для розв'язування задач, проведення віртуальних експериментів в даній предметній галузі.

До СРЗ ставляться наступні вимоги:

- Універсальність використання;
- Переносність;
- Повторне використання коду;
- Стандартизація інтерфейсу взаємодії;
- Інтерфейс користувача повинен бути інтуїтивно зрозумілим;
- Інтерфейс користувача повинен бути максимально наближеним до використання в даній предметній галузі;
- Повинен бути набір команд користувача, за допомогою яких можна робити допустимі перетворення над об'єктами даної предметної галузі.

Розглянемо більш детально програмні вимоги. Універсальність використання та переносність вимагає, щоб розроблений продукт легко було впроваджувати в існуючі програмні середовища, як в локальні версії, так і в версії, які орієнтовані на мережу Інтернет. Повторне використання коду передбачає впровадження продукту без повторної перекомпіляції та без внесення будь-яких змін в коди.

Стандартизація інтерфейсу взаємодії дозволяє вирішити наступні проблеми:

– Незалежність коду від розробника (тобто компоненти може розробляти будь-хто, наслідувавши інтерфейс від існуючого);

– Можливість розробляти програмні засоби під ще не існуюче CP3;

– Легкість локалізації та усунення помилок.

Щоб виконати наведені вимоги, було обрано в якості CP3 елементи управління ActiveX з наперед визначеним інтерфейсом.

Інтерфейс:

```
[
  uuid(8855BFBA-F259-4255-8BF1-89FF93AB2601),
  version(1.0),
  helpstring("LINAOCX1 Proj1 Library"),
  control
]
library LINAOCX1 Proj1
{

  [
    uuid(9DF8C556-DE73-451D-9F6A-614EFFEDA111),
    version(1.0),
    helpstring("Events interface for LINAOCX1 Control")
  ]
  dispinterface ILINAOCX1Events
  {
    properties:
    methods:
    {
      id(0x00000011)
    }
    HRESULT OnSave([in, out] BSTR *xmlstr);
  };

  [
    uuid(68CECF91-85BD-4050-9E84-C1651A0A84BE),
    version(1.0),
    helpstring("LINAOCX1 Control"),
    control
  ]
  coclass LINAOCX1
  {
    [default] interface ILINAOCX1;
    [default, source] dispinterface ILINAOCX1Events;
  };

  [
    uuid(DE58F478-7184-4FD8-A9AF-42FE69CE7A08),
```



```

version(1.0),
helpstring("Dispatch interface for LINAOCX1 Control"),
dual,
oleautomation
]
interface ILINAOCX1: IDispatch
{
    [
        id(0x00000012)
    ]
    HRESULT _stdcall Load([in] BSTR xmlstr );
    [
        propget,
        id(0x00000013)
    ]
    HRESULT _stdcall Task([out, retval] BSTR * Value );
    [
        propput,
        id(0x00000013)
    ]
    HRESULT _stdcall Task([in] BSTR Value );
    [
        id(0x00000014)
    ]
    HRESULT _stdcall Save([out, retval] BSTR * resultXML );
};
};

```

Розглянемо детальніше найважливіші методи, події та властивості, які надає даний інтерфейс.

Методи:

Load – метод призначений для завантаження задачі в CP3;

Save – метод призначений для збереження задачі;

Властивості:

Task – властивість призначена для завантаження та збереження задач;

Події:

OnSave – подія виникає при збереженні задачі.

В якості формату взаємодії обрано XML. Тобто кожна задача є описом її структури мовою розмітки даних XML.

Для того, щоб використати розроблений елемент управління ActiveX CP3 в будь-якій програмі необхідно його створити, потім викликати метод **Load**, куди в якості параметра необхідно передати XML-текст, в якому описується задача. З цього XML тексту середовище повинне повністю відновити ту ситуацію, коли дана задача була збережена.

Коли користувач зберігає задачу, виникає подія **OnSave**. Формат — **OnSave(xmlstr As String)**. В **xmlstr** передається вже сформований XML-текст задачі.

Збереження задачі можна зробити також і з контейнера. Для цього необхідно викликати метод **Save**.

Завантаження та збереження задач можна виконувати за допомогою властивості **Task**. При запису у властивість відбувається завантаження задачі, при читанні — збереження задачі.

При завантаженні задачі до середовища для розв'язування задач необхідно перетворити вхідний формат з мови XML у внутрішній формат даних, який в подальшому буде використовуватися при опрацюванні даних, тобто перетворюватися у процесі розв'язування задач.

Реалізація середовища розв'язування задач для курсу „Лінійна Алгебра”

Середовище для розв'язування задач курсу „Лінійна алгебра” є уніфікованим середовищем для розв'язування задач з лінійної алгебри. Прототипом даного середовища є середовище, розроблене під операційну систему MS-DOS.

Середовище повністю підтримує розв'язування наступних задач над полем раціональних чисел:

- Розв'язати систему лінійних рівнянь;
- Знайти визначник матриці;
- Побудувати обернену матрицю;
- Знайти характеристичний многочлен;
- Знайти власні вектори;
- Знайти ранг матриці;
- Побудувати жорданову форму;
- Ортогоналізувати лінійну оболонку.

Для розв'язування задач передбачено наступні команди:

- Додати дві матриці;
- Помножити матрицю на число;
- Перемножити дві матриці;
- Транспонувати матрицю;
- Одночасно перетворювати матриці.
- Операції над матрицею;
- Відмітити рядок;
- Відмітити стовпчик;
- Додати відмічені рядки;
- Переставити відмічені рядки;
- Переставити відмічені стовпчики;
- Видалити рядок;
- Видалити стовпчик;
- Вставити нульовий рядок;
- Вставити нульовий стовпчик.

В „Середовищі для розв'язування задач” передбачено можливість інтерактивної допомоги – режим роботи „Експерт”. Цей режим вмикається студентом в ситуації, коли він не може продовжити розв'язування задачі. В цьому випадку за програмою оцінюється існуюча ситуація та виконується наступний крок розв'язування. Коли студент повністю розв'язує задачу, експерт перевіряє правильність розв'язування задачі та повідомляє – „Задача не розв'язана”, „Задача розв'язана несамостійно”, „Задача розв'язана самостійно і неправильно”, „Задача розв'язана самостійно і правильно”. Інформація про хід та результати розв'язування задачі надалі зберігаються в зошиті.

В даному середовищі також реалізовано повністю „мишиний інтерфейс”, тобто працювати з середовищем можна, використовуючи лише один маніпулятор „миша”. Наприклад, додавання рядків відбувається „перетягуванням” рядка на рядок, введення множників за допомогою екранної клавіатури.

Застосування „мишиного інтерфейсу” та режиму роботи „Експерт” роблять роботу з середовищем надзвичайно легкою та інтуїтивно зрозумілою.

В середовищі повністю підтримуються принципи компонентно-орієнтованого навчання.

Впровадження середовища в Інтернет-оболонку дозволяє використовувати його як в мережі Інтернет, так і в локальній мережі.

Розроблена модель курсу «Лінійної алгебри» забезпечує:

- виконання державного освітнього стандарту навчання «Лінійної алгебри»;
- поглиблення і розширення теоретичної бази курсу, в першу чергу за рахунок скорочення часу на використання рутинних операцій;
- використання систем віддаленого доступу до структурованого навчального матеріалу для студентів і викладачів, як у синхронному, так і в асинхронному режимах;
- можливість виділення навчальних одиниць, які можуть використовуватися як компоненти розв'язування задач більш високого рівня складності;
- використання комп'ютерно-орієнтованих програм навчального і професійного призначення під час вивчення курсів вищої математики; підвищення практичної значущості результатів навчання;
- створення умов для максимально повного розкриття здібностей студентів, формування необхідного рівня мотивації навчальної діяльності.

ЛІТЕРАТУРА

1. О.В.Співаковський. Підготовка вчителя математики до використання комп'ютера у навчальному процесі. //Комп'ютер у школі та сім'ї.-1999.-№2(6), с. 9-12.

2. Співаковський О.В., Львов М.С., Кравцов Г.М., Крекнін В.А., Гуржій Т.А., Зайцева Т.В., Кушнір Н.А., Кот С.М. Педагогічні технології та педагогічно-орієнтовані програмні системи: предметно-орієнтований підхід. // Комп'ютер у школі та сім'ї. – 2002. – №2 (20). – С. 17–21.
3. Співаковський О.В., В.А. Крекнін. Лінійна алгебра: Навчальний посібник. – Херсон: Айлант, 1997. – 148 с.
4. Співаковський О.В., Крекнін В.А., Черниш К.В. Збірник задач і вправ з лінійної алгебри: Навчальний посібник. – Херсон: Айлант, 2000–206 с.
5. Пейн Крис. Освой самостоятельно ASP.NET за 21 день. : Перевод с англ. – М. : Издательский дом «Вильямс», 2002. -832 с.: ил.
6. Гуннерсон Э. Введение в С#. Библиотека программиста .- СПб: Питер, 2001. -304с. Ил.