

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
«ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ»
АВТОМОБІЛЬНО-ДОРОЖНІЙ ІНСТИТУТ

Факультет «Економіка та управління»
Кафедра «Інформаційні системи в економіці»

ЗАТВЕРДЖУЮ:
Декан факультету

_____ Л. П.
Вовк

Рекомендовано
навчально-методичною
комісією факультету
«Економіка та управління»
протокол засідання № _____

“ _____ ” _____ 20__ р.

від « _____ » _____ 20__ р.

Голова комісії
к.іст.н., доц. _____ М.А. Шипович

РОБОЧА НАВЧАЛЬНА ПРОГРАМА

дисципліни циклу «Дисципліни за вибором навчального закладу»
«Об’єктне програмування»
галузь знань 0305 «Економіка та підприємництво»
напрямок підготовки 6.030502 «Економічна кібернетика»
спеціальність 6.030502 «Економічна кібернетика»

Курс 2 Семестр 3

Рекомендовано кафедрою «Інформаційні системи в економіці»,
протокол № _____ від _____ 20__ р.

Зав.кафедрою
к.т.н., доц.

В. Л. Ніколаєнко

Програму склав
к.т.н., доц.

Д. В. Ніколаєнко

Горлівка 20__ р.

Лист перезатвердження робочої програми
з дисципліни «Об'єктне програмування»

Вніс зміни до програми _____ — « ____ » _____ 20 ____ р.	Рекомендована _____ кафедрою «Інформаційні системи в економіці», протокол засідання № ____ від « ____ » _____ 2008 р., Зав. кафедрою _____ В.Л.Ніколаєнко Затверджена навчально-методичною комісією факультету «Економіка та у п р а в л і н н я » , протокол засідання № ____ від « ____ » _____ 2008 р., Голова комісії _____ М.А.Шипович
Вніс зміни до програми _____ — « ____ » _____ 20 ____ р.	Рекомендована _____ кафедрою «Інформаційні системи в економіці», протокол засідання № ____ від « ____ » _____ 2008 р., Зав. кафедрою _____ В.Л.Ніколаєнко Затверджена навчально-методичною комісією факультету «Економіка та управління», протокол засідання № ____ від « ____ » _____ 2008 р., Голова комісії _____ М.А.Шипович
Вніс зміни до програми _____ — « ____ » _____ 20 ____ р.	Рекомендована _____ кафедрою «Інформаційні системи в економіці», протокол засідання № ____ від « ____ » _____ 2008 р., Зав. кафедрою _____ В.Л.Ніколаєнко Затверджена навчально-методичною комісією факультету «Економіка та управління», протокол засідання № ____ від « ____ » _____ 2008 р., Голова комісії _____ М.А.Шипович

1. ОРГАНІЗАЦІЙНО-МЕТОДИЧНИЙ РОЗДІЛ

1.1. Загальні положення

Робоча програма складена згідно з типовою програмою «Економічна кібернетика», відповідно навчальному плану спеціальності. «Економічна кібернетика», відповідає стандартам Міністерства освіти і науки України (2 0 0 2 р.) підготовки спеціалістів за фахом «Економічна кібернетика», вимог наказу Міністерства освіти України №151 від 02.06.1993 р. і навчальному плану спеціальності 7.030502 «Економічна кібернетика».

На багатьох сучасних підприємствах використовуються комп'ютерні технології, різноманітні технічні засоби, програмне забезпечення, що потребує знань у галузі комп'ютерні науки, тому знання дисципліни необхідно для плідної творчої діяльності сучасного фахівця-бакалавра за спеціальністю «Економічна кібернетика».

Дисципліна складається з таких розділів:

- 1) абстрактні класи;
- 2) властивості класів;
- 3) колекції;
- 4) технологія ADO.NET;
- 5) потоки даних.

1.2. Мета викладання дисципліни

Мета і завдання дисципліни - ознайомити студентів з теоретичними основами принципів об'єктно-орієнтованого програмування – абстракції, інкапсуляції, успадкування, поліморфізму, побудови об'єктних моделей систем, програмних моделей потоків даних, ADO.NET. Здобуття практичних навичок об'єктно-орієнтованого програмування.

1.3. Задачі вивчення дисципліни і основні вимоги до рівня засвоєння змісту дисципліни

Основними задачами вивчення дисципліни є:

- 1) вивчення теоретичних основ і принципів об'єктно - орієнтованого програмування, побудови об'єктних моделей систем;
- 2) вивчення інструментальних засобів;
- 3) засвоєння технологій реалізації абстракцій, поліморфізму, колекцій, ADO.NET;
- 4) розвиток у студентів навичок використання засобів об'єктно-орієнтованого програмування.

В результаті вивчення дисципліни студенти повинні:

- знати:

- 1) засоби реалізації розвинутих властивостей – абстракції, поліморфізму;
- 2) засоби будування колекцій об'єктів;
- 3) засоби роботи з моделлю ADO NET;

- мати навички:

- 1) кодувати різноманітні властивості об'єктів;
- 2) будувати набори об'єктів, реалізовувати принцип поліморфізму;
- 3) використовувати модель і технологію ADO.NET.

1.4. Перелік дисциплін, необхідних для вивчення даної дисципліни

Базою курсу “Об’єктне програмування” є наступні основні дисципліни: “Інформатика та комп’ютерна техніка”.

1.5. Місце дисципліни в професійній підготовці спеціаліста

“Об’єктне програмування” відноситься до циклу „Дисципліни за вибором навчального закладу” і є завершальною при вивченні технологій і засобів розробки програмних моделей систем і, в той же час, є основою при вивченні дисципліни „Об’єктний аналіз і проектування систем”.

2. РОЗКЛАД НАВЧАЛЬНИХ ГОДИН

Розподіл навчальних годин дисципліни “Об’єктне програмування” за основними видами навчальних занять наведено в табл. 2.1.

Таблиця 2.1 - Розклад навчальних годин дисципліни “Об’єктне програмування”

Види навчальних занять	Всього		Семестр
	годин	кредитів ECTS	3
Загальний обсяг дисципліни	180	5	180
– теоретична частина	126	3,5	126
– курсове проектування	54	1,5	54
1. Аудиторні заняття	68		68
з них:	34		34
1.1. Лекції	34		34
1.2. Практичні заняття			
2. Курсове проектування	34		34
з них:			
2.1. Практичні заняття	34		34
3. Самостійна робота	46		46
з них:			
3.1. Опрацювання лекційного матеріалу	17		17
3.2. Підготовка до практичних занять	9		9
3.3. Виконання курсової роботи	20		20
4. Контрольні заходи	32		32

3. ТЕМАТИЧНИЙ ПЛАН

3.1. Семестр 3

3.1.1. Лекційні заняття

Тема і зміст лекцій дисципліни “Об’єктне програмування” наведені в табл. 3.1.

Таблиця 3.1 – Теми і зміст лекцій

Номер Теми	Назва теми та її зміст	Обсяг лекцій, ак. годин	Обсяг самостійної роботи, ак. годин
1	2	3	4
1	Модуль 1. Основні поняття. Предмет та зміст курсу. Зв’язок з іншими дисциплінами. Короткий Зміст, предмет і завдання дисципліни, ресурсне забезпечення.	2	1
2	<u>Абстрактні класи.</u> Абстрактні класи. Поняття абстрактних класів. Ключові слова MustInherit, VustOverride, Overrides. Приклад „Абстрактні класи”.	2	1
3	<u>Колекції об’єктів.</u> Колекції об’єктів Поняття колекції об’єктів. Основні колекції ArrayList, BitArray, DictionaryBase, HashTable, Queu, Stack.	2	1
4	Колекції об’єктів. Ієрархія колекцій. Переваги і недоліки колекції ArrayList. Приклад „ArrayList”.	2	1
5	Колекції об’єктів. Переваги і недоліки колекції HashTable. Приклад „HashTable”.	2	1
6	Колекції об’єктів. Переваги і недоліки колекції CollectionBase. Приклад „CollectionBase”.	2	1
7	<u>Серіалізація і десеріалізація.</u> Серіалізація і десеріалізація об’єктів. Поняття серіалізації, переваги серіалізації над	2	1

	виводом файлів		
	Всього лекційних занять по модулю 1	14	7
1	2	3	4
8	Модуль 2. Сериалізація і десериалізація об'єктів. Переваги і недоліки формату Binary. Приклад „Binary„	2	1
9	<i>Сериалізація і десериалізація об'єктів.</i> Переваги і недоліки формату SOAP. Приклад „SOAP„	2	1
10	<u>Поліморфізм.</u> <i>Поліморфізм.</i> Поняття поліморфізму. Поліморфізм на рівні методів.	2	1
11	<i>Поліморфізм.</i> Приклад розв'язування задачі „Polimorphizm”.	2	1
12	<u>ADO.NET.</u> <i>ADO.NET</i> Модель ADO.NET. Простір імен System.Data.OleDb, System.Data.SqlClient.	2	1
13	<i>ADO.NET</i> Об'єкти: OleDbConnection, SqlConnection, DataAdapter, DataSet. Поняття XML схеми.	2	1
14	<i>ADO.NET</i> Об'єкти: DataTable, DataColumn, DataRow. Властивості: ConnectionString, Provider, DataSource, DataBase.	2	1
15	<u>Потоки даних.</u> <i>Потоки даних.</i> Види потоків даних. Робота з файлами і каталогами.	2	1
16	<i>Потоки даних.</i> Класи Directory, DirectoryInfo, File, FileInfo. Клас Path. Основні атрибути. Клас Directory. Основні атрибути. Приклад “Directory”.	2	1
17	<i>Потоки даних.</i> Клас File. Основні атрибути класу file Приклад розв'язування задачі „DirectoryInfoFileInfo	2	1
	Всього лекційних занять по модулю 2	20	10

3.1.2. Практичні заняття

Таблиця 3.2 – Теми і зміст практичних занять

№ п/п	Назва теми та зміст практичних занять	Обсяг практичних занять, ак. годин	Обсяг самостійної роботи, ак. годин
1	2	3	4
1	Модуль 1. Розробка проекту AbstractClass. Тестування проекту.	6	1
2	Розробка проекту ArrayList. Тестування проекту.	6	1
3	Розробка проекту HashTable. Тестування проекту.	6	2
	Всього практичних занять по модулю 1	18	4
4	Модуль 2. Розробка проекту CollectonBase. Тестування проекту.	4	1
5	Розробка проекту Polimorphizm. Тестування проекту.	6	2
6	Розробка проекту DirectoryInfoFileInfo. Тестування проекту.	6	2
	Всього практичних занять по модулю 2	16	5

3.2. Самостійна робота студентів

Самостійна робота студентів складається з самостійної проробки лекційного матеріалу при підготовці до практичних занять, лекцій, роботи з нормативною, довідковою та періодичною літературою що наведені в табл. 3.1 і 3.2

3.3. Курсове проектування

Виконується курсова робота “Засоби і технології об’єктного програмування”.

Мета курсового проекту:

1) навчити студента навикам використання об'єктних технологій програмування.

2) навчити студента навикам використання об'єктних засобів програмування.

В курсовому проекті на підставі індивідуального завдання необхідно виконати:

- 1) серіалізацію наборів об'єктів Binary;
- 2) серіалізацію наборів об'єктів SOAP;
- 3) серіалізацію наборів об'єктів XML;
- 4) аналіз стандартної об'єктної моделі ADO.NET;
- 5) будівництво власної об'єктної моделі ADO.NET.

Приблизний обсяг пояснювальної записки 30-40 сторінок, проект оболонка, проекти Binary, SOAP, DataGrid.

При виконанні курсової роботи передбачені практичні заняття.

Таблиця 3.3 – Темі і зміст практичних занять з курсового проектування.

№ п/п	Назва теми та зміст практичних занять	Обсяг практичних занять, ак. годин	Обсяг самостійної роботи, ак. годин
1	2	3	4
1	Розробка проекту серіалізація і десеріалізація наборів об'єктів у форматі Binary. Тестування проекту.	10	5
2	Розробка проекту серіалізація і десеріалізація наборів об'єктів у форматі SOAP. Тестування проекту.	10	5
3	Розробка проекту ADO.NET. Тестування проекту.	14	10
	Всього практичних занять	34	20

3.4. Самостійна робота студентів

Під час виконання курсової роботи самостійна робота полягає в роботі з літературою, з MSDN, з примірниками методичних вказівок до виконання курсової роботи, в розробці алгоритмів, передбачених індивідуальним завданням, кодуванні алгоритмів, тестуванні алгоритмів що наведені в таблиці 3.3.

4. ЗАСОБИ ДЛЯ ПРОВЕДЕННЯ ПОТОЧНОГО ТА ПІДСУМКОВОГО КОНТРОЛЮ

4.1. Види контролю

Основні контрольні заходи:

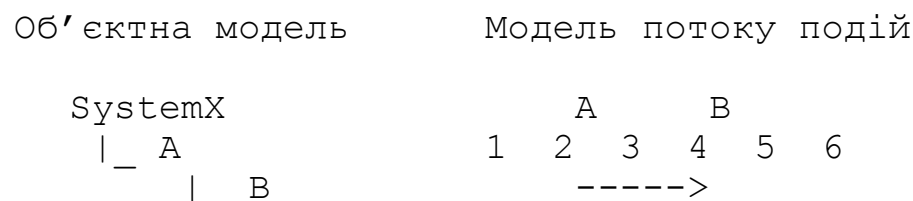
- вхідний (нульовий) контроль;
- поточний;
- модульно-рейтинговий контроль 1;
- модульно-рейтинговий контроль 2;
- підсумковий (семестровий) контроль-іспит;
- контроль знань з вивченої дисципліни.

4.2. Семестр 3

4.2.1. Вхідний контроль

Виконується комплексна контрольна робота по індивідуальним завданням за схемою, що наводиться нижче:

Нехай об'єктний аналіз деякої економічної системи приведе до наведеної об'єктної моделі системи і потоку подій:



Треба:

- 1) побудувати програмну модель об'єктної моделі системи, забезпечивши клас A властивістю ім'я, ініціював його значенням NameA, при розбудові об'єкта екземпляра класу A;
- 2) організувати обробку події від об'єкта A до B.

4.2.2. Перелік типових завдань до 1 модульно-рейтингового контролю знань студентів

1. Поняття абстрактного класу.
2. Структура абстрактного класу.
3. Ключові слова абстрактного класу.
4. Успадкування абстрактного класу.
5. Реалізовані члени абстрактного класу.
6. Не реалізовані члени абстрактного класу.
7. Реалізація членів абстрактного класу у класах.

8. Поняття колекції.
9. Доступ до елементів колекції по ключу.
10. Доступ до елементів колекції по індексу.
11. Приклади стандартних колекцій.
12. Колекції як розширення поняття масив.
13. Гетерогенні колекції.
14. Ієрархія стандартних колекцій.
15. Об'єктна модель стандартних колекцій.
16. Основні атрибути колекцій.
17. Основні властивості стандартних колекцій
18. Основні методи стандартних колекцій
19. Організація власних колекцій на підставі ArrayList.
20. Організація власних колекцій на підставі HashTable
21. Організація власних колекцій на підставі CollectionBase
22. Ітерації по елементах колекцій.
23. Оператор While – EndWhile.
24. Додання елементу в колекцію.
25. Вилучення елементу з колекції.
26. Рахування елементів колекції.
27. Серіалізація об'єктів. Binary.
28. Серіалізація об'єктів. SOAP.
29. Серіалізація об'єктів. XML.
30. Десеріалізація об'єктів.
31. Що розуміється під терміном «.NET Framework»?
32. Чи залежать програми, що розробляються в .NET, від платформи?
33. Що розуміється під зоною видимості змінної в мові VB .NET
34. Як позначається зона видимості змінної в мові
35. Як співвідноситься час життя змінної і зона видимості?
36. Що розуміється під терміном «простір імен»?
37. Що розуміється під терміном «клас»?
38. Які елементи визначаються у складі класу?
39. Яке співвідношення понять «клас» і «об'єкт»?
40. Що розуміється під терміном «члени класу»?
41. Що розуміється під терміном «конструктор»?
42. Скільки конструкторів може містити клас мови VB .NET
43. Приведіть синтаксис опису класу в спільному вигляді.
Проілюструйте його фрагментом програми на мові VB .NET
44. Які модифікатори типа доступу Вам відомі?
45. У чому полягають особливості доступу членів класу з модифікатором public?

46. У чому полягають особливості доступу членів класу з модифікатором `private`?
47. У чому полягають особливості доступу членів класу з модифікатором `protected`?
48. У чому полягає призначення конструктора?
49. Чи кожен клас мови C# має конструктор?
50. Які умовчання для конструкторів прийняті в мові VB .NET
51. Що розуміється під терміном «деструктор»?
52. У чому полягає призначення деструкції?
53. Що розуміється під терміном «спадкоємство»?
54. Що спільного має дочірній клас з батьківським?
55. Що розуміється під терміном «інтерфейс»?
56. Чим відрізняється синтаксис інтерфейсу від синтаксису абстрактного класу?
57. Що розуміється під терміном «событие»?
58. Чи є події членами класів?
59. Яке ключове слово мови VB .NET використовується для опису подій?
60. Приведіть синтаксис опису події в спільному вигляді.
Проілюструйте його фрагментом програми на мові VB .NET
61. Які оператори мови VB .NET використовуються для обробки виключень?
62. У чому полягає значення механізму виключень?
63. Приведіть синтаксис блоку `try...catch` в спільному вигляді
64. чи Залежить виклик блоку `finally` від наявності виключення?
65. Що таке об'єктна модель?
66. Перерахуйте недоліки процедурного і модульного програмування.
67. Перерахуйте кроки, необхідні для створення екземпляра класу.
68. Поясніть кроки, необхідні для створення екземпляра класу.
69. Яка різниця між списком аргументів і списком параметрів?
70. Як звертатися до змінних екземпляра з програми?
71. Що таке спадкоємство?
72. Яким чином використання інкапсуляції допомагає досягти цілей об'єктно-орієнтованого програмування?
73. Що таке перевантаження методу?
74. Функція: передача параметрів і повернення значень при виклику.
Прототип функції.
75. Передача параметрів у функцію за значенням і по засланню.

4.2.3. Перелік типових завдань до 2 модульно-рейтингового контролю знань студентів

1. Основна ідея ООП. Класи і об'єкти. Розділи класу. Інтерфейс класу.
2. Перехід до ООП як етап розвитку технології розробки Пп.
3. Основні поняття ООП. Клас і об'єкт.
4. Принципи ООП. Інкапсуляція. Спадкоємство. Поліморфізм.
5. Отримання елементу колекції.
6. Перелік елементів колекції.
7. Вилучення елементу з колекції.
8. Поняття специфікатору Default
9. Властивості і методи по умовчання.
10. Поняття поліморфізму.
11. Реалізація поліморфізму при роботі з колекціями.
12. Недоліки СУБД.
13. Шляхи подолання недоліків СУБД.
14. Суть технології ADO.NET.
15. Конструювання класів. Контроль доступу до елементів класу, розділи класу private і public.
16. Конструктори і деструкції.
17. Правила привласнення. Сумісність типів для об'єктів.
18. Приклад на створення, привласнення і знищення динамічних об'єктів.
19. Зберігання даних в списку. Додавання і видалення вузлів в лінійному списку.
20. Черга, стік.
21. Проблема зберігання об'єктів в декількох списках одночасно.
22. Об'єктна модель ADO.NET.
23. Поняття XML схеми.
24. Поняття XML мови програмування.
25. Поняття джерела даних.
26. Види джерел даних.
27. Поняття провайдера даних.
28. Види провайдерів даних.
29. Поняття віртуального набору даних.
30. Об'єктна модель провайдера даних.
31. Об'єктна модель віртуального набору даних.
32. Основні методи і властивості провайдера даних.
33. Основні методи і властивості віртуального набору даних.
34. Засоби будування зв'язку з наборами даних.
35. Програмний спосіб будування зв'язку, провайдера, набору даних.
36. Додавання звичайного стовпця.
37. Додавання ключового стовпця.

38. Додавання авто стовпця.
39. Додавання рахує мого стовпця.
40. Додавання строки даних.
41. Додавання строки даних циклічне.
42. Абстрактні класи
43. Поняття колекції. Колекція Arraylist
44. Поняття колекції. Колекція Hashtable
45. Поняття колекції. Колекція Collectionbase
46. Поняття потоку даних і види потоків
47. Поняття потоку даних. Потік Binary
48. Поняття потоку даних. Потік SOAP
49. Поняття поліморфізму
50. ADO.NET. Основні поняття
51. ADO.NET. Об'єктна модель
52. ADO.NET. Поняття провайдера даних. Види провайдерів.
Ключові об'єкти провайдерів
53. ADO.NET. Поняття XML схеми
54. ADO.NET. Поняття з'єднання. Способи створення з'єднань
55. ADO.NET. Поняття з'єднання. Способи створення з'єднань.
56. ADO.NET. Загальний вид властивості Connection
57. ADO.NET. Схема використання Dataadapter
58. ADO.NET. Способи створення Dataadapter
59. ADO.NET. Основні атрибути Dataadapter
60. ADO.NET. Dataset. Просте додавання стовпців
61. ADO.NET. Dataset. Додавання автозростаючого стовпця
62. ADO.NET. Dataset. Додавання обчислюваного стовпця
63. ADO.NET. Dataset. Додавання стовпця первинного ключа
64. ADO.NET. Dataset. Додавання рядків просто
65. ADO.NET. Dataset. Додавання рядків в масиві
66. ADO.NET. Dataset. Додавання рядків в циклі
67. ADO.NET. Уведення-виведення. Локальне зберігання даних
68. ADO.NET. Уведення-виведення. Клас Directory
69. ADO.NET. Уведення-виведення. DirectoryInfo, File Info
70. ADO.NET. Уведення-виведення. Приклад Recurstrree
71. ADO.NET. Уведення-виведення. Основні атрибути класу Stream
72. ADO.NET. Уведення-виведення. Файловий потік. Байтовий обмін
73. ADO.NET. Уведення-виведення. Файловий потік. Обмін
примітивними типами даних
74. ADO.NET. Уведення-виведення. Файловий потік. Обмін
текстовими даними

75.ADO.NET. Уведення-виведення. Файловий потік. Способи створення об'єктів Streamreader, Streamwriter

4.2.4. Підсумковий (семестровий) контроль-іспит

До семестрового контролю-іспиту винесені питання 1 і 2 модульно-рейтингового контролю знань, а також практичні задачі по організації абстрактних класів, поліморфізму, колекцій.

4.2.5. Контроль знань з вивченої дисципліни

Виконується комплексна контрольна робота по індивідуальним завданням:

Нехай об'єктний аналіз деякої економічної системи приводе до наведеної об'єктної моделі системи і потоку подій:

Об'єктна модель

```

SystemX
|_ A
    |_ B

```

Модель потоку подій

```

      A      B
1  2  3  4  5  6
----->

```

Треба:

- 1) побудувати програмну модель об'єктної моделі системи, забезпечивши клас A властивістю ім'я, ініціював його значенням NameA, при розбудові об'єкта екземпляра класу A;
- 2) організувати обробку події від об'єкта A до B;
- 3) клас SystemX розглядати як абстрактний з реалізованою властивістю Name і MustOverrides методом Salary;
- 4) функціональність класу A - алгоритм методу Salary - імітувати оператором MsgBox.

5. ПЕРЕЛІК НАВЧАЛЬНО-МЕТОДИЧНОЇ ЛІТЕРАТУРИ І НАВЧАЛЬНИХ ЗАСОБІВ

5.1. Основна та додаткова література

Основна:

1. Корнелл Г., Моррисон Дж. Программирование в VB.NET: учебный курс.- СПб.: Питер, 2002. - 400с.: ил.
2. Рохилла С., Натан С., Милхотра С. Microsoft ADO.NET: разработка профессиональных проектов. - СПб.:БХВ – Петербург, 2003. - 788с.: ил.
3. Visual Basic .NET. Искусство программирования Пер. с англ./ Пол Киммел – СПб: ООО «Диасофт ЮП», 2003.- 720с.
4. Вильдермьюс, Шон. Практическое использование ADO.NET. Доступ к данным в Internet.: /Пер. с англ. - М.: Издательский дом «Вильямс», 2003- 288с.: ил.

Додаткова:

1. Сайлер, Брайан, Споттс, Джефф. Использование . Специальное издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 752 с.: ил.
2. Франклин, Кит. VB.NET для разработчиков.: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 272 с.: ил.
3. Рихтер Дж. Программирование на платформе Microsoft .NET Framework. «Русская редакция», 2002.
4. Gilmore S. Programming in Standard ML '97: A Tutorial Introduction. The University of Edinburgh, 2897.
5. Harper R. Programming in Standard ML. Carnegie Mellon University, 2001.
6. Керниган Б., Пайк Р. Практика программирования.–СПб.: Невский диалект, 2001.–381 с.
7. Браунси К. Основные концепции структур данных и реализация в C++.– М.: Издательский дом «Вильямс», 2002.– 320 с.
8. Вольфенгаген В.Э. Конструкции языков программирования. Приемы описания. М.: АО «ЮрИнфоР», 2001, 276 с.

5.2. Методичні посібники і вказівки

1. Методичні вказівки до виконання курсової роботи з дисципліни „Об’єктне програмування”. Эл. Форма.
2. Методичні вказівки до виконання "Лабораторного практикуму-1" з дисципліни"Інформатика і комп’ютерна техніка” для студентів

спеціальності „Економічна кібернетика” / Горлівка: АДІ ДонНТУ, 2001. - 204 с.

3. Методичні вказівки до виконання завдань лабораторного практикуму 2 та 3. Эл. Форма.