


МІЖРЕГІОНАЛЬНА  
АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ



МАУП



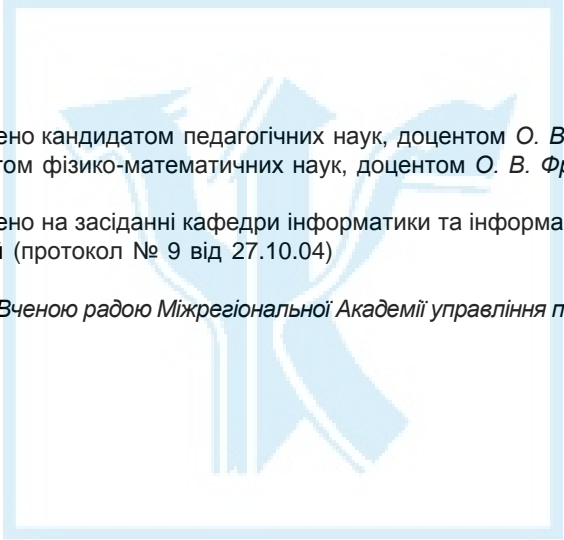
**НАВЧАЛЬНА ПРОГРАМА  
дисципліни**

**“МОВИ ОБ’ЄКТНО-ОРІЄНТОВАНОГО  
ПРОГРАМУВАННЯ”**

**(для бакалаврів)**

МАУП

Київ 2005



Підготовлено кандидатом педагогічних наук, доцентом *О. В. Вітюком*  
і кандидатом фізико-математичних наук, доцентом *О. В. Франчуком*

Затверджено на засіданні кафедри інформатики та інформаційних  
технологій (протокол № 9 від 27.10.04)

*Схвалено Вченою радою Міжрегіональної Академії управління персоналом*

**Вітюк О. В., Франчук О. В.** Навчальна програма дисципліни “Мови об’єктно-орієнтованого програмування” (для бакалаврів). — К.: МАУП, 2005. — 12 с.

Навчальна програма містить пояснювальну записку, навчально-тематичний план, програмний матеріал до вивчення дисципліни “Мови об’єктно-орієнтованого програмування”, теми практичних занять, контрольні питання, орієнтовні теми курсових проектів, а також список рекомендованої літератури.

## **ПОЯСНЮВАЛЬНА ЗАПИСКА**

Мета навчальної програми дисципліни “Мови об’єктно-орієнтованого програмування” — ознайомити студентів з концепцією об’єктно-орієнтованого програмування (ООП) з використанням мов ООП Inprise Delphi, Borland C++ або Microsoft Visual C++.

Курс розрахований на студентів з певним досвідом програмування мовою Pascal, C.

**Завдання** дисципліни: ознайомити студентів з основними поняттями мов ООП як сучасного інструментарію зі створення об’єктно-орієнтованого програмного забезпечення.

На початку курсу наводиться короткий огляд структурного програмування і розглядаються його відмінності від ООП, вивчаються основні (базові) поняття ООП. Далі основна увага приділяється тим засобам і конструкціям вибраної мови ООП, які являють собою відповідну мову програмування високого рівня (Pascal або C). Програма курсу містить особливості вибраної мови ООП, що не мають еквівалента у відповідній базовій мові програмування. Наприкінці курсу обговорюються проблеми, що виникають при використанні мов ООП для проектування і розробки великих програмних систем.

### **НАВЧАЛЬНО-ТЕМАТИЧНИЙ ПЛАН вивчення дисципліни**

#### **“МОВИ ОБ’ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ”**

№ пор.	Назва теми
1	Вступ. Парадигми і мови програмування
2	Програмне середовище розробника
3	Базові поняття програмування мови ООП
4	Процедурно-орієнтоване програмування
5	Об’єктне програмування
6	Об’єктно-орієнтоване програмування
7	Проблеми розробки програмних систем

**ПРОГРАМНИЙ МАТЕРІАЛ**  
**до вивчення дисципліни**

**“МОВИ ОБ’ЄКТНО-ОРІЄНТОВАНОГО**  
**ПРОГРАМУВАННЯ”**

**Тема 1. Вступ. Парадигми і мови програмування**

Процедурне програмування (Фортран, Алгол, Паскаль, С).  
Об’єктне програмування (Модула 2), об’єктно-орієнтоване програмування (Delphi, C++, Java).

*Література* [2; 6; 8; 13]

**Тема 2. Програмне середовище розробника**

Препроцесор, компілятор. Директиви препроцесора. Програмний модуль.

*Література* [5; 15; 18–20; 24]

**Тема 3. Базові поняття програмування мови ООП**

Літерали, їх типи. Сталі та змінні, ініціалізація змінних, присвоєння, зведення типів при присвоєнні.

Вказівники (pointer), виділення та звільнення пам’яті.

Вирази, оператори (операції), зведення типів у виразах.

Інструкції: тотожна, розгалуження, вибір, цикли з лічильником, за умовою повторення (закінчення), передача управління: (break, continue).

Масиви. Літерні величини.

Записи, поля.

*Література* [2; 3; 6; 13; 15; 20; 24; 26]

**Тема 4. Процедурно-орієнтоване програмування**

Функції: визначення і оголошення, вбудовані функції. Типи формальних параметрів і тип обчисленого значення. Передача параметрів значенням, особливості передачі параметрів у разі

вказівників, масивів; замовчувані значення параметрів; перетворення типів при передачі параметрів. Обчислення значення функції — вихід із функції; особливості повернення та використання значення.

Рекурсія.

Видимість: глобальна область видимості, локальна область видимості (функція, блок), визначення і оголошення глобальних об'єктів і функцій.

Тривалість життя об'єктів: автоматичні, статичні і динамічні об'єкти, оператори управління пам'яттю.

*Література* [2; 6–8; 13–15; 17–20; 24–26]

### **Тема 5. Об'єктне програмування**

Принцип інкапсуляції даних. Конструктори, деструктори.

Класи і об'єкти, атрибути об'єктів — члени класів, функції класу, функції поза класом, область видимості і права доступу.

Інтерфейс класу, реалізація класу; визначення і оголошення класу; вбудовані функції класу. Ідентичність об'єктів, копіювання, ініціалізація і присвоєння, ліквідація об'єктів.

Поточний об'єкт `self`.

Створення та ініціалізація об'єктів, властивості конструкторів, довизначення конструкторів, замовчуваний конструктор, обмеження прав доступу до конструктора. Ліквідація об'єктів, властивості деструкторів.

Довизначення (*overloading*) операторів (операцій-функцій): арифметичні операції (бінарні, унарні, суміщені з присвоєнням), оператори управління пам'яттю, оператор виклику функції, оператор доступу до елемента масиву, оператор доступу за указником, оператор присвоєння, конвертори (перетворення типів).

*Література* [3; 9; 17; 22; 27]

### **Тема 6. Об'єктно-орієнтоване програмування**

Ієрархія об'єктів і ієрархія класів, базовий клас, дочірній клас, успадкування, уточнення привілеїв доступу до членів ба-

зового класу, захищена область класу, область видимості і оператор розв'язування області видимості.

Створення дочірніх об'єктів, конструктор дочірнього класу, послідовність виклику конструкторів, ліквідація дочірніх об'єктів, деструктор дочірнього класу, послідовність виклику деструкторів.

Статичне і динамічне зв'язування: поліморфізм, віртуальні функції, заміщення функцій, віртуальні деструктори.

*Література* [1; 3; 9; 10; 17; 22; 27]

### **Тема 7. Проблеми розробки програмних систем**

Програма, програмний комплекс, програмний продукт, комплексний програмний продукт; ознаки складних систем.

Життєвий цикл програмного забезпечення. Фази розробки програмного забезпечення: початкова, уточнення, планування, власне розробка (аналіз, проектування, програмування, тестування), приймання та приймальні випробування, супровід.

Поняття про об'єктно-орієнтоване програмування, проектування та аналіз. Об'єктна модель програмного забезпечення. Елементи об'єктної моделі: абстрагування, інкапсуляція, модульність, ієрархічність. Переваги об'єктної моделі.

*Література* [1; 4–6; 8; 9; 11–13; 21; 23; 25]

### **ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ**

1. Літерали, їх типи. Сталі та змінні, ініціалізація змінних, присвоєння, зведення типів при присвоєнні.
2. Вказівники (pointer), виділення та звільнення пам'яті. Вирази, оператори (операції), зведення типів у виразах. Інструкції: тотожна, розгалуження, вибір, цикли з лічильником, за умовою повторення (закінчення), передача управління (break, continue).
3. Масиви. Літерні величини. Записи, поля.
4. Функції: визначення і оголошення, вбудовані функції. Типи формальних параметрів і тип обчисленого значення.

Передача параметрів значенням, масивів; замовчувані значення параметрів; перетворення типів при передачі параметрів.

5. Обчислення значення функції — вихід із функції; особливості повернення та використання значення. Рекурсія.
6. Видимість: глобальна область видимості, локальна область видимості, визначення і оголошення глобальних об'єктів і функцій.
7. Тривалість життя об'єктів: автоматичні, статичні і динамічні об'єкти, оператори управління пам'яттю (`new`, `dispose`).
8. Принцип інкапсуляції даних. Конструктори, деструктори. Класи і об'єкти, атрибути об'єктів — члени класів, функції класу, область видимості і права доступу.
9. Інтерфейс класу; реалізація класу; визначення і оголошення класу; вбудовані функції класу. Ідентичність об'єктів, копіювання, ініціалізація і присвоєння, ліквідація об'єктів. Поточний об'єкт `self`.
10. Створення та ініціалізація об'єктів, властивості конструкторів, довизначення конструкторів, обмеження прав доступу до конструктора. Ліквідація об'єктів, властивості деструкторів.
11. Розробка алгоритмів з використанням об'єктів.
12. Ієрархія об'єктів і ієрархія класів, базовий клас, дочірній клас, успадкування, уточнення привілеїв доступу до членів базового класу, захищена область класу, область видимості.
13. Створення дочірніх об'єктів, конструктор дочірнього класу, послідовність виклику конструкторів, ліквідація дочірніх об'єктів, деструктор дочірнього класу, послідовність виклику деструкторів.
14. Статичне і динамічне зв'язування: поліморфізм, віртуальні функції, заміщення функцій, віртуальні деструктори.
15. Розробка алгоритмів з використанням класів.

## **КОНТРОЛЬНІ ПИТАННЯ**

### **До теми 1**

1. Охарактеризуйте мови програмування Фортран, Паскаль, С та сфери їх застосування.
2. Відмінності між процедурним, об'єктним та об'єктно-орієнтованим програмуванням.

### **До теми 2**

1. Що таке препроцесор?
2. Що таке компілятор? Особливості компілятора середовища візуального програмування.
3. Що таке директиви препроцесора? Яке їх призначення?
4. Що називають програмним модулем?

### **До теми 3**

1. Опишіть способи ініціалізації змінних різних типів.
2. Як виділити та звільнити область пам'яті для вказівника (pointer)?
3. Особливості синтаксису реалізації лінійних структур, розгалуження, вибору та циклів.
4. Як оголосити масив? Опишіть основні операції обробки масивів.
5. Як оголосити літерну змінну? Основні можливості обробки.
6. Що таке запис? Для чого використовуються записи?

### **До теми 4**

1. Опишіть призначення функцій та особливості їх оголошення.
2. Охарактеризуйте особливості передачі параметрів функції значенням, масивів; замовчувані значення параметрів.
3. Особливості повернення та використання значення функцій.
4. Що таке рекурсія? Наведіть приклади рекурсивних алгоритмів.
5. Области видимості функцій.
6. Визначення і оголошення глобальних об'єктів і функцій.
7. Тривалість життя об'єктів: автоматичні, статичні і динамічні об'єкти.



## До теми 5

1. Принцип інкапсуляції даних.
2. Конструктори, деструктори.
3. Класи і об'єкти, атрибути об'єктів — члени класів.
4. Функції класу, функції поза класом.
5. Область видимості і права доступу до функцій класу.
6. Інтерфейс класу, реалізація класу.
7. Визначення і оголошення класу; вбудовані функції класу.
8. Ідентичність об'єктів, копіювання, ініціалізація і присвоєння об'єктів.
9. Поточний об'єкт `self`.
10. Особливості створення та ініціалізація об'єктів.
11. Властивості конструкторів, довизначення конструкторів, замовчуваній конструктор.
12. Ліквідація об'єктів, властивості деструкторів.

## До теми 6

1. Ієрархія об'єктів і ієрархія класів.
2. Що називають базовим класом?
3. Що називають дочірнім класом?
4. Що називають успадкуванням?
5. Захищена область класу.
6. Область видимості класу.
7. Особливості створення та ліквідації дочірніх об'єктів.
8. Послідовність виклику конструкторів.
9. Деструктор дочірнього класу, послідовність виклику деструкторів.
10. Охарактеризуйте статичне і динамічне зв'язування.
11. Що таке віртуальні функції та віртуальні деструктори?

## До теми 7

1. Що таке поліморфізм?
2. Назвіть ознаки складних систем.
3. Опишіть фази розробки програмного забезпечення.
4. Охарактеризуйте елемент об'єктної моделі — абстрагування.
5. Охарактеризуйте елемент об'єктної моделі — інкапсуляцію.
6. Охарактеризуйте елемент об'єктної моделі — модульність.

7. Охарактеризуйте елемент об'єктної моделі — ієрархічність.
8. У чому переваги об'єктної моделі над процедурною?

## **ОРІЄНТОВНІ ТЕМИ КУРСОВИХ ПРОЕКТІВ**

Курсовий проект передбачає розробку студентом у вибраній системі ООП (Delphi або C++) повноцінного програмного засобу (ПЗ), що використовує графічний інтерфейс системи Windows, а також відповідної супровідної документації (довідки, описання можливостей та використаних алгоритмів тощо). Розроблена програма (залежно від типу) повинна дозволяти завантажувати та зберігати результати роботи у стандартних графічних або текстових форматах файлів системи.

Орієнтовні типи програмних засобів:

1. Графічний редактор (зразок — MSPaint).
2. Текстовий редактор (зразок — WordPad).
3. Адресна книга (зразок — “Адресная книга”).
4. Програма обміну повідомленнями у локальній мережі (зразок — “NetMeeting”).
5. Файл-менеджер (зразок — “Проводник”).
6. Калькулятор (зразок — “Калькулятор”).
7. Універсальний програвач (зразок — “Универсальный проигрыватель”).

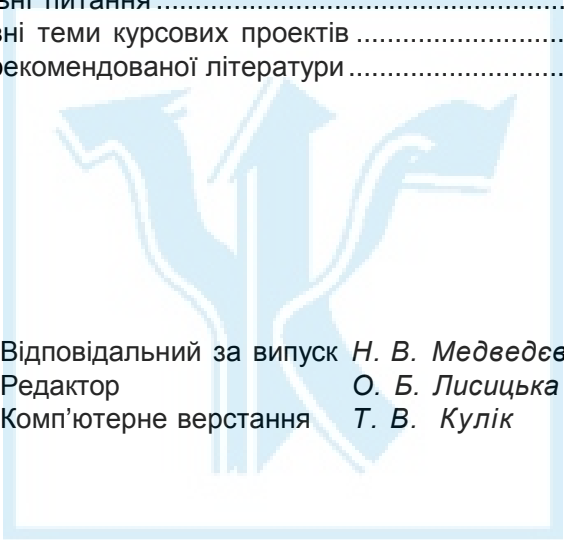
## **СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ**

1. *Delphi*. Online Documentation.
2. Ахо Ал. В., Дж. Э. Хопкрофт, Дж. Д. Ульман. Структуры данных и алгоритмы. — М.: Вильямс, 2000. — 384 с.
3. Антон Элиенс. Принципы объектно-ориентированной разработки программ. — М.: Вильямс, 2000. — 496 с.
4. Ахо А., Хопкрофт Дж., Дж Ульман. Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979. — 540 с.
5. Брукс Ф. Как проектируются и создаются программные комплексы. — М.: Наука, 1979.
6. Вирт Н. Алгоритмы + структуры данных = программы. — М.: Мир, 1985. — 406 с.
7. Грогано П. Программирование на языке Паскаль. — М.: Мир, 1982. — 384 с.

8. *Гудман С., Хидетниemi С.* Введение в разработку и анализ алгоритмов — М.: Мир, 1981. — 366 с.
9. *Макгрегор Джон, Сайкс Давид.* Тестирование объектно-ориентированного программного обеспечения: Практ. пособие. — М.: ТИД ДС, 2002. — 432 с.
10. *Дарахвелидзе П. Г., Марков Е. П.* Delphi — среда визуального программирования. — СПб.: BHV, 1996. — 352 с.
11. *Зелковиц М., Шоу А., Геннон Дж.* Принципы разработки программного обеспечения. — М.: Мир, 1982. — 368 с.
12. *Соммервилл Иан.* Инженерия программного обеспечения. — М.: Вильямс, 2002. — 624 с.
13. *Кнут Д.* Искусство программирования для ЭВМ. Основные алгоритмы. — М.: Мир, 1976. — 735 с.
14. *Липпман С. Б., Лажоие Ж.* Язык программирования С++: Вводный курс. — М.: ДМК, 2001.
15. *Марченко А. И., Марченко Л. А.* Программирование в среде Turbo Pascal 7.0. — К.: ВЕК+Ю, 2000. — 464 с.
16. *Матчо Дж., Фолкнер Р. Ф.* DELPHI. — М.: БИНОМ, 1995.
17. *Павловская Т. А.* С/С++ Программирование на языке высокого уровня. — СПб.: Питер, 2001.
18. *Семашко Г. Л., Салтыков А. И.* Программирование на языке Паскаль, — М.: Наука, 1988. — 122 с.
19. *Сердюченко В. Я.* Розробка алгоритмів та програмування мовою Turbo Pascal. — Харків: Паритет, 1995. — 352 с.
20. *Ставровський А.* Турбо Паскаль 7.0.: Учебник. — К.: BHV, 2000. — 400 с.
21. *Канер Сэм, Фолк Джек, Нгуен Енг Кек.* Тестирование программного обеспечения. — ДиаСофт, 2000. — 544 с.
22. *Миллер Т., Пауэл Д.* Использование Delphi 3: Спец. изд. — К.; М.: Диалектика, 1997.
23. *Уокер Ройс.* Управление проектами по созданию программного обеспечения. — Лори, 2002. — 448 с.
24. *Фаронов В. В.* Турбо Паскаль 7.0. Начальный курс: Учеб. пособие. — М.: Нолидж, 2001. — 576 с.
25. *Фокс Дж.* Программное обеспечение и его разработка. — М.: КОМПЬЮТЕР ПРЕСС, 1995. — 280 с.
26. *Шилдт Г.* Теория и практика С++. — СПб.: BHV, 1996.
27. *Янг М.* Visual C++ 6. — К.: BHV, 1999.

## **ЗМІСТ**

Пояснювальна записка .....	3
Навчально-тематичний план вивчення дисципліни “Мови об’єктно-орієнтованого програмування” .....	3
Програмний матеріал до вивчення дисципліни “Мови об’єктно-орієнтованого програмування” .....	4
Теми практичних занять .....	6
Контрольні питання .....	8
Орієнтовні теми курсових проектів .....	10
Список рекомендованої літератури .....	10



Відповідальний за випуск *Н. В. Медведєва*  
Редактор *О. Б. Лисицька*  
Комп’ютерне верстання *Т. В. Кулік*

**МАУП**  
Зам. № ВКЦ-2011

Міжрегіональна Академія управління персоналом (МАУП)  
03039 Київ-39, вул. Фрометівська, 2, МАУП