

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
Національний авіаційний університет
Факультет комп'ютерних систем
Кафедра комп'ютеризованих систем управління

ЗАТВЕРДЖУЮ

Ректор

_____ М.Кулик
"_____" _____ 2012р.



Система менеджменту якості

НАВЧАЛЬНА ПРОГРАМА
навчальної дисципліни
"Об'єктно-орієнтоване програмування"
(за кредитно-модульною системою)

Галузь знань: 0501 "Інформатика та обчислювальна техніка"
Напрямок підготовки: 6.050102 "Комп'ютерна інженерія"

Курс – 4 Семестр – 8

Лекції – 24 Екзамен – 8 семестр

Лабораторні заняття – 24

Самостійна робота – 78

Усього (годин/кредитів ECTS) – 126/3,5

Курсова робота – 8 семестр

Індекс Р4-6.050102-2/12-4.3

СМЯ НАУ НІ 09.02.02-01-2012



Навчальна програма дисципліни "Об'єктно-орієнтоване програмування" розроблена на основі освітньо-професійної програми та навчального плану № НБ-4-6.050102-2/12 підготовки фахівців освітньо-кваліфікаційного рівня "Бакалавр" за напрямом 6.050102 "Комп'ютерна інженерія", "Тимчасового Положення про організацію навчального процесу за кредитно-модульною системою (в умовах педагогічного експерименту)" та "Тимчасового Положення про рейтингову систему оцінювання", затверджених наказом ректора від 15.06.2004 №122/од, та наказу ректора від 12.04.2005 №81/од.

Навчальну програму розробили
професор кафедри комп'ютеризованих

систем управління _____ О.Коба

доцент кафедри комп'ютеризованих

систем управління _____ С.Пустова

Навчальна програма обговорена та схвалена на засіданні випускової кафедри напряму 6.050102 "Комп'ютерна інженерія" (спеціальність 7/8.05010202 "Системне програмування") – кафедри комп'ютеризованих систем управління, протокол № _____ від " _____ " _____ 2012 р.

Завідувач кафедри _____ О.Литвиненко


Навчальна програма обговорена та схвалена на засіданні науково-методично-редакційної комісії факультету комп'ютерних систем, протокол № _____ від " _____ " _____ 2012 р.

Голова НМРК _____ Б.Масловський

Рівень документа – 3б

Плановий термін між ревізіями – 1 рік

Контрольний примірник

	Система менеджменту якості. Навчальна програма навчальної дисципліни "Об'єктно-орієнтоване програмування"	Шифр документа	СМЯ НАУ НП 09.02.02 – 01-2012
		Стор. 3 із 10	

ЗМІСТ

	стор.
1. Пояснювальна записка	4
1.1. Місце навчальної дисципліни в системі професійної підготовки фахівця .	4
1.2. Мета викладання навчальної дисципліни.....	4
1.3. Завдання вивчення навчальної дисципліни.....	4
1.4. Інтегровані вимоги до знань та умінь з навчальної дисципліни	4
1.5. Інтегровані вимоги до знань і умінь з навчальних модулів.....	4
1.6. Міждисциплінарні зв'язки навчальної дисципліни.....	6
2. Зміст навчальної дисципліни	6
2.1. Модуль №1 "Побудова АТД і об'єктів".	6
2.2. Модуль №2 "Поліморфізм і розробка ОО програми".	7
2.3. Модуль №3 "Курсова робота".	7
3. Список рекомендованих джерел.....	8
3.1. Основні рекомендовані джерела	8
3.2. Додаткові рекомендовані джерела	8
4. Форми документів Системи менеджменту якості	9



1. ПОЯСНЮВАЛЬНА ЗАПИСКА

1.1. Місце навчальної дисципліни в системі професійної підготовки фахівця

Дана навчальна дисципліна є основою сукупності знань та вмінь, що формують у фахівця в області системного програмування навички конструювання сучасних об'єктно-орієнтованих (ОО) програмних систем.

1.2. Мета викладання навчальної дисципліни

Метою викладання дисципліни є проектування, розробка та кодування програмних систем за допомогою засобів об'єктно-орієнтованого програмування (ООП) для розв'язку задач системного програмування в технічних та телекомунікаційних системах.

1.3. Завдання вивчення навчальної дисципліни

Завданнями вивчення навчальної дисципліни є:

- конструювання (проектування) програмних систем на основі відповідного набору абстрактних типів даних (АТД);
- проектування в термінах взаємозв'язку АТД та застосування механізму успадкування для сумісного використання коду та інтерфейсу;
- використання віртуальних функцій для динамічної обробки зв'язаних об'єктів;
- проектування конкретних прикладних задач з використанням шаблонів та власних розробок;
- користування програмними засобами, розробленими за допомогою ООП-методології.

1.4. Інтегровані вимоги до знань та умінь з навчальної дисципліни

У результаті вивчення даної навчальної дисципліни студент повинен:

Знати:

- основні концепції та ключові моменти ООП – парадигми.

Вміти:

- самостійно розробляти алгоритми об'єктно-орієнтованих програм з використанням мов програмування високого рівня;
- самостійно проектувати, розробляти, відлагоджувати та використовувати програмні додатки на основі об'єктно-орієнтованого підходу.

1.5. Інтегровані вимоги до знань і умінь з навчальних модулів

Навчальний матеріал дисципліни структурований за модульним принципом і складається з двох класичних навчальних модулів. Окремим третім модулем є курсова робота, яка виконується у восьмому семестрі.

1.5.1. У результаті засвоєння навчального матеріалу навчального модуля №1 "Побудова АТД і об'єктів" студент повинен:

Знати:

- поняття класів та об'єктів;
- дві концепції ООП: інкапсуляцію і успадкування;



– практичні засоби підтримання інкапсуляції, успадкування, статичного поліморфізму.

Вміти:

– самостійно розробляти алгоритми, кодувати та відлагоджувати програмні додатки з використанням концепцій інкапсуляції, успадкування та статичного поліморфізму;

– самостійно схематизувати предметну область проекту, використовуючи концепцію АТД;

– самостійно організовувати взаємозв'язки типів, використовуючи ієрархію класів;

– самостійно практично застосовувати концепцію статичного поліморфізму.

1.5.2. У результаті засвоєння навчального матеріалу навчального модуля №2 "Поліморфізм і розробка ОО програми" студент повинен:

Знати:

– концепцію поліморфізму;

– практичні засоби ОО мовою програмування підтримки динамічного і параметричного поліморфізму і розробки ОО програми (віртуальні функції, родові функції родові класи, шаблони, контейнери, ітератори, виняткові ситуації);

– особливості системи вводу/виводу.

Вміти:

– практично використовувати концепцію динамічного і параметричного поліморфізму;

– створювати власні родові функції і родові класи (шаблони);

– використовувати бібліотеку стандартних шаблонів;

- проектувати, розробляти і кодувати мовою ООП – програмні додатки, комбінуючи власні розробки та стандартні шаблони і функції.

1.5.3. У результаті виконання курсової роботи (модуль №3) студент повинен:

Знати:

- поняття класів і об'єктів;

- три концепції ООП: інкапсуляція, успадкування, поліморфізм – і практичні засоби їх реалізації;

- шаблони та їх реалізацію;

- організацію системи вводу/виводу в ОО програмних системах.

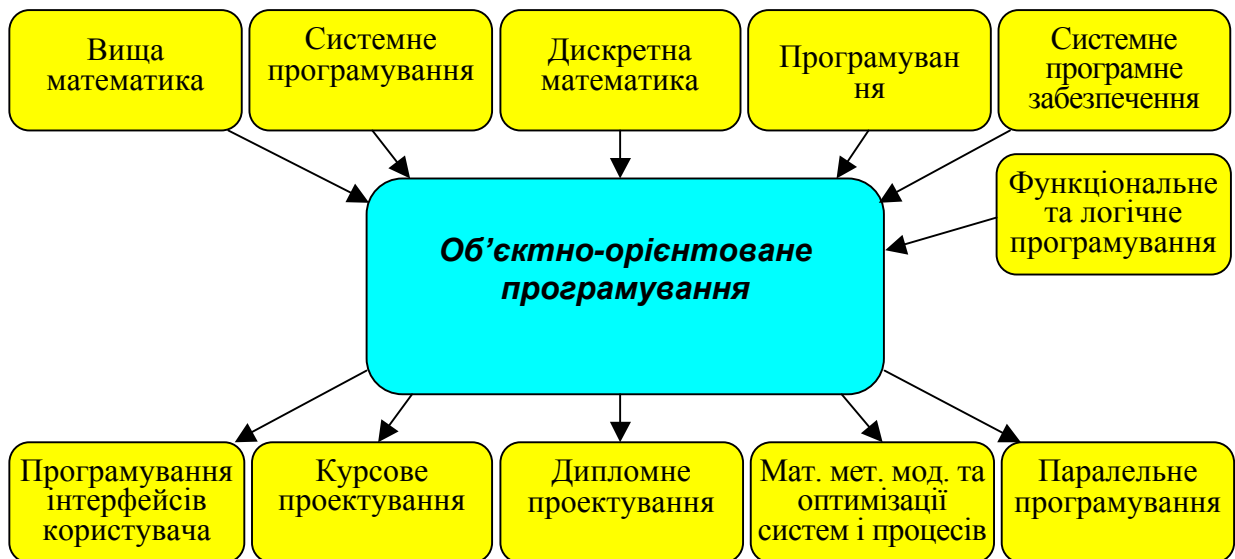
Вміти:

- практично застосовувати набуті знання з ОО-методології і технології проектування для власного проекту і відповідного програмного модуля ОО мовою.

Знання та вміння, отримані студентом під час вивчення даної навчальної дисципліни, використовуються в подальшому при вивченні багатьох наступних дисциплін професійної підготовки фахівця з базовою та повною вищою освітою.



1.6. Міждисциплінарні зв'язки навчальної дисципліни



2. ЗМІСТ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

2.1. Модуль №1 "Побудова АТД і об'єктів".

Тема 2.1.1. ООП – сучасна методологія програмування.

Еволюція мов програмування. ООП – підхід в програмуванні. Інкапсуляція, успадкування, поліморфізм. Принципи проектування даних. Принципи роботи і використання ОО–систем.

Тема 2.1.2. Класи і об'єкти.

Класи. Відкриті і закриті члени класу. Об'єкти. Конструктори. Деструктори. Конструктори з параметрами. Присвоєння об'єктів. Передача об'єктів функціям. Об'єкти як значення, що повертаються функціями. Вбудовані функції в оголошенні класу.

Тема 2.1.3. Масиви об'єктів, вказівники і посилання.

Масиви об'єктів. Вказівники на об'єкти. Вказівник this. Оператори new і delete. Посилання. Передача посилань на об'єкти. Посилання як значення, що повертаються функцією. Незалежні посилання.

Тема 2.1.4. Статичний поліморфізм. Перевантаження функцій.

Дружні функції. Перевантаження функцій. Перевантаження конструкторів. Конструктори копій. Аргументи за замовчуванням. Перевантаження та неоднозначність.

Тема 2.1.5. Статичний поліморфізм. Перевантаження операторів.

Перевантаження унарних і бінарних операторів. Перевантаження операторів відношення. Перевантаження логічних операторів. Перевантаження



унарних операторів. Дружні оператор-функції. Особливості використання оператора присвоєння. Перевантаження оператора індекса масиву [].

Тема 2.1.6. Успадкування.

Принцип успадкування. Специфікатори доступу при успадкуванні. Управління доступом до базового класу. Захищенні члени класу. Конструктори і деструктори при успадкуванні. Множинне успадкування. Віртуальні базові класи. Проектування програмного забезпечення і успадкування.

2.2. Модуль №2 "Поліморфізм і розробка ОО програми".

Тема 2.2.1. Динамічний поліморфізм. Віртуальні функції.

Віртуальні функції. Вказівники на похідні класи. Абстрактні класи. Чисто віртуальні функції. Застосування віртуальних функцій.

Тема 2.2.2. Параметричний поліморфізм. Шаблони.

Родові функції. Родові класи. Шаблони. Застосування родових функцій і родових класів. Шаблони і успадкування. Шаблони і дружні функції.

Тема 2.2.3. Стандартна бібліотека шаблонів.

Шаблони контейнерних класів. Контейнери. Вектор. Список. Дек. Множина і мультимножина. Бітові множини. Відображення і мультивідображення. Стек. Черга. Пріоритетна черга.

Ітератори. Типи ітераторів. Методи роботи з ітераторами. Функції і об'єкти функцій.

Строкові шаблони. Виклик строкових шаблонів. Перетворення в строки. Порівняння строк. Пошук в строках.

Алгоритми.

Тема 2.2.4. Оброблення виняткових ситуацій.

Поняття виняткової ситуації. Генерація виняткових ситуацій. Перехоплення виняткових ситуацій. Оброблення виняткових ситуацій, що виникають під час застосування оператора виділення пам'яті. Конструктори, деструктори і оброблення виняткових ситуацій. Виняткові ситуації та успадкування.

Тема 2.2.5. Система введення/виведення.

Базові положення системи введення/виведення. Форматне введення/виведення. Файлове введення/виведення. Маніпулятори введення/виведення. Функції введення користувача. Функції виведення користувача. Функції вставки і вилучення.

2.3. Модуль №3 "Курсова робота".

Курсова робота (КР) з дисципліни виконується у восьмому семестрі відповідно до затверджених в установленому порядку методичних рекомендацій



з метою закріплення та поглиблення теоретичних знань та вмінь, набутих студентом у процесі засвоєння навчального матеріалу дисципліни в області об'єктно-орієнтованого програмування.

Виконання КР є важливим етапом у підготовці до виконання дипломного проекту (роботи) майбутнього фахівця з системного програмування.

Конкретна мета КР полягає у розробці програмного проекту об'єктно-орієнтованою мовою програмування відповідно до індивідуального завдання, заданого в методичних рекомендаціях з курсового проектування, із застосуванням принципів і методів ООП.

3. СПИСОК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

3.1. Основні рекомендовані джерела

3.1.1. Шилдт Г. Самоучитель C++: Пер. с англ.. — 3-е изд. — СПб. : BHV - Санкт-Петербург, 2006. — 688с.

3.1.2. Шилдт Г. Полный справочник по C++ — 4.изд. — М. : СПб.; К. : Издательский дом "Вильямс", 2003. — 796с.

3.1.3. Х.М.Дейтел, П.Дж.Дейтел. Как программировать на C++. – 5-е изд. – М. : Бином-Пресс, 2008. – 1456 с.

3.1.4. Прата С. Язык программирования C++. – 5-е изд. – М. ; СПб. ; К. : Вильямс, 2007. – 1184 с.

3.2. Додаткові рекомендовані джерела

3.2.1. Павловская Т.А., Щупак Ю.А. C/C++. Структурное программирование: Практикум. – СПб. : Питер, 2005. – 239 с.

3.2.2. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. – М.: Бином, 2000. – 560 с.

