

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»
ННК «ІПСА»**

ЗАТВЕРДЖУЮ
Заступник директора ННК
«ІПСА»

_____ В.Д. Романенко
(підпис) (ініціали, прізвище)

« ____ » _____ 2014 р.

Об'єктно-орієнтоване програмування
(назва навчальної дисципліни)

**ПРОГРАМА
навчальної дисципліни**

підготовки **бакалаврів**
(назва освітньо-кваліфікаційного рівня)

напряму **6.050101 Комп'ютерні науки**
(шифр і назва)

(шифр за ОПП – 1.3.1.02)

Ухвалено методичною комісією
ННК «ІПСА»
(назва інституту/факультету)

Протокол від _____ 2014 р. № ____

Голова методичної комісії

_____ П.І. Бідюк
(підпис) (ініціали, прізвище)

« ____ » _____ 2014 р.

Київ – 2014

РОЗРОБНИКИ ПРОГРАМИ:

Асистент, к.т.н. Булах Богдан Вікторович

(підпис)

(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові)

Доцент, к.т.н. Гемба Ольга Володимирівна

(підпис)

(посада, науковий ступінь, вчене звання, прізвище, ім'я, по батькові)

Програму затверджено на засіданні кафедри системного проектування
Протокол від 12 листопада 2014 року № 3

Завідувач кафедри

А.І. Петренко
(ініціали, прізвище)

(підпис)

12 листопада 2014 р.

© НТУУ «КПІ», 2014 рік

I. ЗАГАЛЬНІ ВІДОМОСТІ

Об'єктно-орієнтоване програмування (ООП) є однією з сучасних парадигм програмування і дозволяє спростити процес розробки складного програмного забезпечення за допомогою об'єктної декомпозиції. ООП базується на представленні програми у вигляді сукупності взаємодіючих об'єктів, кожен з яких є екземпляром певного класу, що може бути пов'язаний з іншими класами у ієрархію наслідування. Об'єктно-орієнтований підхід підтримується чималим числом мов програмування та засобів автоматизації розробки програмного забезпечення.

Дисципліна «Об'єктно-орієнтоване програмування» відноситься до циклу професійної та практичної підготовки; базується на знанні дисципліни «Програмування та алгоритмічні мови» та забезпечує вивчення курсів «Теорія мов програмування», «Технологія створення програмних продуктів», «Проектування інформаційних систем».

Метою дисципліни є теоретичне вивчення основ об'єктно-орієнтованої парадигми програмування та отримання практичних навичок у розробці програм згідно принципів об'єктно-орієнтованої технології розробки програмних систем.

Завдання вивчення дисципліни полягає у формуванні системи наступних знань та умінь.

Знання:

- базових концепцій та принципів ООП (об'єкт, клас, інкапсуляція, ієрархія, наслідування, поліморфізм);
- підходів до проектування об'єктно-орієнтованих систем, основних шаблонів проектування;
- принципів узагальненого програмування.

Уміння:

- практично застосовувати принципи ООП при розробці програмних систем: проводити об'єктну декомпозицію системи, проектувати ієрархію класів, доречно використовуючи типові шаблони проектування;
- розробляти комп'ютерні програми згідно поставлених задач за допомогою сучасних об'єктно-орієнтованих мов програмування.

Форма навчання	Семестри	Всього кредитів/годин	Розподіл навчального часу за видами занять						Семестрова атестація
			Лекції	Практичні заняття	Семінарські заняття	Лабораторні роботи	Комп'ютерний практикум	СРС	
Денна	3	6/216	54			36		126	екз.

II. ЗМІСТ НАВЧАЛЬНОГО МАТЕРІАЛУ

Розділ 1. Принципи та концепції об'єктно-орієнтованого програмування

Тема 1.1. Основні концепції ООП. Класи та об'єкти.

Вступ. Мета, основні задачі та зміст дисципліни. Парадигми програмування.

Визначення, основні поняття та терміни ООП. Мова програмування C++. Клас та

об'єкт. Опис інтерфейсу та реалізації класів. Реалізація інкапсуляції у вигляді класу. Приховання даних. Специфікатори доступу та поняття області видимості. Простори імен. Вкладені класи. Дружні функції та класи. Конструктори, їх призначення та типи. Деструктори. Константні та статичні поля та методи. Перевантаження методів та операторів. Операції як члени класу та як дружні функції.

Тема 1.2. Ієрархія класів

Агрегація та композиція класів. Створення, ініціалізація та знищення вкладених об'єктів. Наслідування, його типи. Базовий клас. Специфікатори доступу при наслідуванні. Приведення типів. Конструктори та деструктори в похідних класах. Множинне наслідування. Поліморфізм. Віртуальні функції. Таблиця віртуальних функцій. Статичне та динамічне зв'язування. Віртуальні деструктори.

Обробка виняткових ситуацій. Виняток (exception). Загальна схема обробки виняткових ситуацій. Генерація та перехоплення винятку. Виняток у конструкторі та деструкторі. Стандартні винятки.

Підтримка принципів ООП в різних мовах програмування. Засоби узагальненого опису об'єктно-орієнтованих систем. Мови узагальненого опису ОО-систем, UML-діаграм. Діаграми класів. Автоматична генерація коду.

Розділ 2. Узагальнене програмування

Тема 2.1. Шаблони функцій та класів

Шаблони функцій та класів. Обмеження шаблонів C++. Шаблони, наслідування, дружні функції та класи.

Тема 2.2. Стандартна бібліотека шаблонів

Стандартна бібліотека шаблонів C++ (STL). Контейнери послідовностей. Асоціативні контейнери. Ітератори та доступ до елементів контейнерів. Алгоритми STL. Алгоритми порівняння, пошуку, сортування. Математичні алгоритми. Адаптери контейнерів. Функціональні об'єкти (функтори).

Розділ 3. Принципи проектування об'єктно-орієнтованих систем

Тема 3.1. Шаблони проектування

Класифікація шаблонів (патернів) проектування. Вирішення проблем за допомогою проектування. Правила вибору шаблонів. Шаблони створення. Структурні шаблони. Шаблони поведінки.

Розділ 4. Окремі аспекти розробки об'єктно-орієнтованих систем

Тема 4.1. Програмування систем на подіях

Події та повідомлення. Механізм подій та його підтримка в різних мовах програмування. Системи передачі повідомлень. Підходи до реалізації в C++.

Тема 4.2. Розробка систем з графічним інтерфейсом користувача

Графічний інтерфейс користувача (GUI). Віконні програми: загальні принципи, засоби розробки.

Тема 4.3. Продуктивність розробки та виконання об'єктно-орієнтованих програм

Розробка об'єктно-орієнтованих програм для різних платформ. Ефективність виконання об'єктно-орієнтованих програм. Розширення до стандарту C++.

III. ПРИБЛИЗНА ТЕМАТИКА ПРАКТИЧНИХ (СЕМІНАРСЬКИХ) ЗАНЯТЬ

Практичні та семінарські заняття за навчальним планом відсутні.

IV. ПРИБЛИЗНИЙ ПЕРЕЛІК ЛАБОРАТОРНИХ РОБІТ

Метою курсу лабораторних робіт є закріплення теоретичного матеріалу та отримання студентами практичних навичок з використання принципів об'єктно-орієнтованого програмування при розробці програмного забезпечення. Перелік лабораторних робіт:

1. Розробка складних типів даних. Перевантаження операцій.
2. Розробка ієрархії класів. Наслідування та поліморфізм.
3. Використання стандартної бібліотеки шаблонів.

V. ІНДИВІДУАЛЬНІ СЕМЕСТРОВІ ЗАВДАННЯ

Індивідуальні семестрові завдання за навчальним планом відсутні.

VI. КОНТРОЛЬНІ РОБОТИ

Для проведення модульного контролю передбачена одна модульна контрольна робота з розділів 1-2 за темами 1.1, 1.2, 2.1, 2.2.

Мета контрольної роботи – перевірка засвоєння студентами основних принципів об'єктно-орієнтованого програмування та узагальненого програмування, практичних навичок з програмування на об'єктно-орієнтованій мові C++.

VII. МЕТОДИЧНІ ВКАЗІВКИ

При вивченні матеріалу курсу рекомендується особливу увагу приділити закріпленню базових положень, поступово ускладнюючи матеріал із наведенням відповідних прикладів коду на об'єктно-орієнтованій мові програмування. Доцільним є використання ілюстративного або мультимедійного матеріалу, що забезпечує асоціативне запам'ятовування.

Перші лекції теми 1.1 дозволяють пов'язати викладення матеріалу із попередньо пройденим курсом «Програмування та алгоритмічні мови» і забезпечити цілісне сприйняття матеріалу та його узгодженість із здобутими раніше знаннями та навичками. Розділи 3 та 4 забезпечують підґрунтя для подальшого вивчення курсів «Технологія створення програмних продуктів» та «Проектування інформаційних систем».

VIII. НАВЧАЛЬНО-МЕТОДИЧНІ МАТЕРІАЛИ

Основна література

1. Дейтел П. Д., Дейтел Х. М. Как программировать на C++. 3 изд. – М.: Бином, 2003. – 1152 с.
2. Давыдов В.Г. Технологии программирования. C++. – СПб.: БХВ-Петербург, 2005. — 672 с.
3. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования. – СПб: Питер, 2001. – 368 с.
4. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на

- C++ . 2-е изд. / Пер. с англ. – М.: «Издательство Бином», СПб: «Невский диалект», 1998.
5. Лафоре Р. Объектно-ориентированное программирование в C++. — СПб.: Питер. — 2004. — 928 с.
6. Страуструп Б. Язык программирования C++. Спец.изд. — М.: Бином. — 2004. — 1054 с.

Допоміжна література

7. Саттер Г. Решение сложных задач на C++. -- М.: “Вильямс”, 2002. -- 400с.
8. Саттер Г. Новые сложные задачи на C++. -- М.: “Вильямс”, 2008. -- 272 с.
9. Майерс С. Наиболее эффективное использование C++. 35 новых способов улучшить стиль программирования. -- СПб.: Питер, 2006. -- 296с.
10. Рихтер Дж. CLR via C#. Программирование на платформе Microsoft.NET Framework 2.0 на языке C#. Мастер-класс./Пер. с англ. – 2-е изд., исправ. –М.: Издательство «Русская редакция», СПб.: Питер, 2008. – 656с.
11. Найгел К., Ивсен Б., Глинн Дж., Скиннер М., Уотсон К. C# 2005 и платформа .NET 3.0 для профессионалов.: Пер. с англ. – М.: ООО «И.Д. Вильямс», 2008. – 1376с.
12. Хорстманн К.С., Корнелл Г. Java 2. Библиотека профессионала — 8-е изд. — М.: Вильямс, 2008. — Т. I: Основы. — 816 с.

Навчальна програма складена на основі освітньо-професійної програми СВО

напрям: 6.050101 – «Комп’ютерні науки», спеціальність: 8.05010102 – «Інформаційні технології проектування»

(шифр та назва напряму, спеціальності)

Розробник(и) програми: **к.т.н., доц. Гемба О.В., Булах Б.В.**

(вчений ступінь, звання, прізвище та ініціали автора (-ів) програми)