

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»  
ННК «ІПСА»**

ЗАТВЕРДЖУЮ  
Заступник директора ННК  
«ІПСА»

\_\_\_\_\_ В.Д. Романенко  
(підпис) (ініціали, прізвище)

« \_\_\_\_ » \_\_\_\_\_ 2014 р.

\_\_\_\_\_ В.Д. Романенко  
(підпис) (ініціали, прізвище)

« \_\_\_\_ » \_\_\_\_\_ 2015 р.

**1.3.1.02 Об'єктно-орієнтоване програмування**  
(назва навчальної дисципліни)

**РОБОЧА ПРОГРАМА  
кредитного модуля**

**підготовки**

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

**напряму 6.050101 Комп'ютерні науки**

(шифр і назва)

**форми навчання**

**денної**  
(денна/заочна)

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

Робоча програма кредитного модуля Об'єктно-орієнтоване програмування  
для студентів за напрямом підготовки 6.050101 Комп'ютерні науки,  
освітньо-кваліфікаційного рівня БАКАЛАВР, за денною формою навчання  
складена відповідно до програми навчальної дисципліни Об'єктно-орієнтоване  
програмування.

Розробники робочої 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)

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

Завідувач кафедри

\_\_\_\_\_  
(підпис)

А.І. Петренко

(ініціали, прізвище)

12 листопада 2014 р.

© НТУУ «КПІ», 2014 рік  
© НТУУ «КПІ», 2015 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».

Загальна кількість кредитів ETCS складає 6, номер за переліком дисциплін програми підготовки: НП-03.

## II. РОЗПОДІЛ НАВЧАЛЬНОГО ЧАСУ

Семестр / код кредитного модуля	Всього годин	Розподіл годин за видами занять							Кількість МКР	Вид індивідуального завдання	Семестрова атестація
		Лекції	Практичні заняття	Семінарські заняття	Лабораторні роботи	Комп'ютерний практикум	СРС				
							Всього	У тому числі на виконання індивідуального завдання			
3/НП-03	216	54	–	–	36	–	126	126	1	Курсова робота	Екзамен

## III. МЕТА І ЗАВДАННЯ КРЕДИТНОГО МОДУЛЯ

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.

## IV. ТЕМАТИЧНИЙ ПЛАН

### IV.1. РОЗПОДІЛ НАВЧАЛЬНОГО ЧАСУ ЗА ТЕМАМИ

Назви розділів, тем	Розподіл по видах занять						
	Всього	Лекції	Практичні заняття (контрольні роботи)	Семінарські заняття	Лабораторні роботи	Комп'ютерний практикум	СРС
<b>Розділ 1. Принципи та концепції об'єктно-орієнтованого програмування</b>	<b>84</b>	<b>26</b>			<b>24</b>		<b>34</b>
Тема 1.1. Основні концепції ООП. Класи та об'єкти.	44	14			12		18
Тема 1.2. Ієрархія класів	40	12			12		16
<b>Розділ 2. Узагальнене програмування</b>	<b>42</b>	<b>10</b>			<b>12</b>		<b>20</b>
Тема 2.1. Шаблони функцій та класів	16	2			6		8
Тема 2.2. Стандартна бібліотека шаблонів	21	7			6		8
Контрольна робота по розділам 1,2	5	1					4
<b>Розділ 3. Принципи проектування об'єктно-орієнтованих систем</b>	<b>16</b>	<b>8</b>					<b>8</b>
Тема 3.1. Шаблони проектування	16	8					8
<b>Розділ 4. Окремі аспекти розробки об'єктно-орієнтованих систем</b>	<b>20</b>	<b>10</b>					<b>10</b>
Тема 4.1. Програмування систем на подіях	4	2					2
Тема 4.2. Розробка систем з графічним інтерфейсом користувача	4	2					2
Тема 4.3. Продуктивність розробки та виконання об'єктно-орієнтованих програм	12	6					6
<b>Курсова робота</b>	<b>18</b>						<b>18</b>
<b>Екзамен</b>	<b>36</b>						<b>36</b>
<b>Разом</b>	<b>216</b>	<b>54</b>	<b>-</b>	<b>-</b>	<b>36</b>	<b>-</b>	<b>126</b>

## ***IV.2. ЛЕКЦІЇ***

### **Розділ 1, Тема 1.1**

#### **Лекція 1. Принципи та концепції об'єктно-орієнтованого програмування**

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

Визначення, основні поняття та терміни ООП.

Рекомендована література: [4] гл.1

#### **Лекція 2. Об'єктно-орієнтовані мови програмування**

Критерії об'єктної орієнтованості мови програмування. Огляд мов програмування, що підтримують об'єктно-орієнтовану парадигму. Мова програмування C++: основні конструкції, відмінності від процедурної мови С.

Рекомендована література: [4] гл.1, [1] гл.2-5, [5] гл.2-5

#### **Лекція 3. Клас та об'єкт.**

Синтаксична конструкція class. Опис інтерфейсу та реалізації класів. Створення об'єктів класу. Динамічне керування виділенням пам'яті для об'єктів.

Рекомендована література: [1] гл.6, [5] гл.6, [6] гл.10

#### **Лекція 4. Інкапсуляція та приховання даних**

Реалізація інкапсуляції у вигляді класу. Приховання даних. Специфікатори доступу та поняття області видимості. Простори імен. Вкладені класи. Дружні функції та класи.

Рекомендована література: [1] гл.6-7, [5] гл.6, [6] гл.10

#### **Лекція 5. Ініціалізація та знищення об'єктів**

Конструктори, їх призначення. Конструктор за замовчанням, конструктор копіювання, конструктор приведення. Деструктори, їх призначення.

Рекомендована література: [1] гл.6, [5] гл.6, [6] гл.10

#### **Лекція 6. Константні та статичні елементи класу**

Константні поля та методи, особливості використання. Статичні поля та методи класу, особливості використання.

Рекомендована література: [1] гл.7, [5] гл.6, [6] гл.10

#### **Лекція 7. Перевантаження операцій**

Сигнатура методу. Перевантаження методів класів. Перевантаження операторів: синтаксис, обмеження. Операції як члени класу та як дружні функції.

Рекомендована література: [1] гл.8, [5] гл.8, [6] гл.11

### **Розділ 1, Тема 1.2**

#### **Лекція 8. Агрегація та композиція класів**

Агрегація та композиція як спосіб повторного використання коду. Об'єкти, посилання та вказівники на них як поля класів. Створення, ініціалізація та знищення вкладених об'єктів.

Рекомендована література: [1] гл.7, [5] гл.6, [6] гл.10

#### Лекція 9. Наслідування

Типи наслідування. Базовий клас. Специфікатори доступу при наслідуванні. Приведення типів. Конструктори та деструктори в похідних класах. Множинне наслідування.

Рекомендована література: [1] гл.9, [5] гл.9

#### Лекція 10. Віртуальні функції та поліморфізм

Поліморфізм. Віртуальні функції. Абстрактні базові класи. Таблиця віртуальних функцій. Статичне та динамічне зв'язування. Віртуальні деструктори.

Рекомендована література: [1] гл.10, [5] гл.11

#### Лекція 11. Обробка виняткових ситуацій

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

Рекомендована література: [1] гл.13, [6] гл.14

#### Лекція 12. Підтримка принципів ООП в різних мовах програмування

Реалізація принципів ООП у мовах C# та Java та принципові відмінності від мови програмування C++.

Рекомендована література: [10], [11], [12]

#### Лекція 13. Засоби узагальненого опису об'єктно-орієнтованих систем

Мови узагальненого опису ОО-систем, UML-діаграми. Діаграми класів. Автоматична генерація коду.

Рекомендована література: [1]

### Розділ 2, Тема 2.1

#### Лекція 14. Шаблони функцій та класів

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

Рекомендована література: [1] гл.13, [5] гл.14

### Розділ 2, Тема 2.2

#### Лекція 15. Стандартна бібліотека шаблонів

Стандартна бібліотека шаблонів C++ (STL). Основні елементи бібліотеки. Приклади використання.

Рекомендована література: [1] гл.20, [5] гл.15

#### Лекція 16. Контейнери та ітератори

Контейнери послідовностей. Асоціативні контейнери. Ітератори та доступ до елементів контейнерів.

Рекомендована література: [1] гл.20, [5] гл.15

#### Лекція 17. Алгоритми

Алгоритми STL. Алгоритми порівняння, пошуку, сортування. Математичні алгоритми.

Рекомендована література: [1] гл.20, [5] гл.15

#### Лекція 18. Адаптери та функтори

Адаптери контейнерів. Функціональні об'єкти (функтори). Приклади використання.

Рекомендована література: [1] гл.20, [5] гл.15

### Розділ 3, Тема 3.1

#### Лекція 19. Огляд шаблонів проектування

Класифікація шаблонів (патернів) проектування. Вирішення проблем за допомогою проектування. Правила вибору шаблонів.

Рекомендована література: [3]

#### Лекція 20. Шаблони створення

Абстрактна фабрика (Abstract Factory). Будівельник (Builder). Фабричний метод (Factory Method). Прототип (Prototype). Єдиний екземпляр (Singleton).

Рекомендована література: [3]

#### Лекція 21. Структурні шаблони

Адаптер (Adapter). Міст (Bridge). Компонувальник (Composite). Декоратор (Decorator). Фасад (Facade). Пристосуванець (Flyweight). Заступник (Proxy)

Рекомендована література: [3]

#### Лекція 22. Шаблони поведінки

Ланцюжок відповідальності (Chain of Responsibility). Команда (Command). Інтерпретатор (Interpreter). Ітератор (Iterator). Посередник (Mediator). Хранитель (Memento). Спостерігач (Observer). Відвідувач (Visitor)

Рекомендована література: [3]

### Розділ 4, Тема 4.1

#### Лекція 23. Події та повідомлення

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

Рекомендована література: [4]

### Розділ 4, Тема 4.2

#### Лекція 24. Системи з графічним інтерфейсом користувача

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

Рекомендована література: [4]

### Розділ 4, Тема 4.3

#### Лекція 25. Розробка об'єктно-орієнтованих програм для різних платформ

Проблема кросплатформності розробки. Особливості розробки під ОС Linux та Windows. Підходи до розробки мультиплатформних програм.

#### Лекція 26. Ефективність виконання об'єктно-орієнтованих програм

Вплив механізмів реалізації концепції ООП на продуктивність виконання програм. Підходи до зменшення накладних витрат.

Рекомендована література: [1], [5], [6]

Лекція 27. Огляд розширень до стандарту С++

Стандарти С++ та їх розвиток. Бібліотеки розширених можливостей. Огляд засобів бібліотеки Boost.

Рекомендована література: [1] гл.21

### ***IV.3. ПРАКТИЧНІ ЗАНЯТТЯ***

Практичні заняття за навчальним планом відсутні.

### ***IV.4. СЕМІНАРСЬКІ ЗАНЯТТЯ***

Семінарські заняття за навчальним планом відсутні.

### ***IV.5. ЛАБОРАТОРНІ РОБОТИ (КОМП'ЮТЕРНИЙ ПРАКТИКУМ)***

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

№ ЛР	Назва лабораторної роботи	Розділ	Тема	Год. (СРС)
1.	Розробка складних типів даних. Перевантаження операцій	1	1.1	12 (4)
2.	Розробка ієрархії класів. Наслідування та поліморфізм	1	1.2	12 (4)
3.	Використання стандартної бібліотеки шаблонів	2	2.2	12 (4)
Всього	3 роботи			36 (12)

### ***IV.6. ІНДИВІДУАЛЬНІ ЗАВДАННЯ***

Індивідуальні семестрові завдання за навчальним планом відсутні.

### ***IV.7. 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 С++.

## ***V. 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».

Рейтинг студента з дисципліни складається з балів, що він отримує за усі види робіт. Приблизний внесок робіт у підсумковий бал:

- 1) виконання та захист 3 лабораторних робіт – 30%;
- 2) виконання модульної контрольної роботи – 10%;
- 3) виконання курсової роботи – 20%;
- 4) відповідь на екзамені – 40%.

Додатково пропонуються штрафні та заохочувальні бали. Штрафні бали надаються за несвоєчасну здачу лабораторних робіт та курсової роботи без поважних причин. Заохочувальні бали надаються за удосконалення дидактичних матеріалів з дисципліни, роботу на лекціях, вдалі оригінальні рішення при виконанні лабораторних робіт.

Положення про рейтингову систему наведено у «Додатку А».

## **VI. 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
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.

Робоча навчальна програма складена на основі навчальної програми  
дисципліни: «Об'єктно-орієнтоване програмування»  
(назва дисципліни)

затвердженої

(посада і дата затвердження)

Розробник(и) програми

к.т.н., доц. Гемба О.В,  
Булах Б.В.

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

\_\_\_\_\_/\_\_\_\_\_  
(підпис) (прізвище та ініціали)

## ДОДАТОК А

### ПОЛОЖЕННЯ

#### про рейтингову систему оцінки успішності студентів

з кредитного модуля (дисципліни)

НП-03 Об'єктно-орієнтоване програмування

(код та назва)

для спеціальності:

8.05010102 Інформаційні технології проектування

(шифр та назва)

ННК «ІПСА»

Розподіл навчального часу за видами занять і завдань з дисципліни згідно з робочим навчальним планом.

Семестр	Навчальний час		Розподіл навчальних годин				Контрольні заходи		
	кредити	акад. рік	Лекц.	Прак-Тичні	Лаб. роб.	СРС + Екз.	МКР	РГР	Семестр атест.
3	6	216	54	–	36	126	1	–	екз.

Рейтинг студента з дисципліни складається з балів, що він отримує за:

- 1) виконання однієї модульної контрольної роботи;
- 2) виконання та захист 3 лабораторних робіт;
- 3) виконання та захист курсової роботи;
- 4) відповідь на екзамені.

#### *Система рейтингових (вагових) балів та критерії оцінювання*

##### 1. Модульна контрольна робота

Ваговий бал – 15.

Критерії оцінювання кожного завдання:

Позначення	Бали, %	Опис
+	100	Задачу вирішено вірно
+ <sub>-</sub>	75-99	Є дрібні недоліки в вирішенні або оформленні
±	50-74	Більша частина задачі вирішена, ідея вирішення вірна, однак у вирішенні є помилки або вирішення не доведене до кінця
±	30-49	Хід вирішення вірний, але зроблені грубі помилки, що не дають можливості отримати вірний результат
- <sub>+</sub>	1-29	Хід вирішення намічено, але зроблені критичні помилки
–	0	Задачу не вирішено або вирішено невірно

##### 2. Лабораторні роботи

Максимальна кількість балів за всі лабораторні роботи дорівнює 45 балам. Критерії оцінювання з визначенням рівнів за кожний етап (виконання лабораторної роботи, якість захисту роботи):

Лаб. роб.	№1	№2	№3
Виконання	5	5	5
Захист	10	10	10

Загальний бал	15	15	15
Термін виконання	1-5 заняття	6-10 заняття	11-15 заняття

### Штрафні та заохочувальні бали

Вид діяльності	Бали
Захист лабораторної роботи або МКР пізніше встановленого терміну без поважної причини	-2 бали за кожне заняття пізніше встановленого терміну
Участь у модернізації лабораторних робіт (оригінальні рішення при її виконанні), виконання завдань із удосконалення дидактичних матеріалів;	+2-6 балів
Активна робота на лекціях.	+1-6 балів

### **Розрахунок шкали (R) рейтингу:**

Сума вагових балів контрольних заходів протягом семестру складає (стартовий рейтинг):

$$R_C = 15 + 45 = 60 \text{ балів}$$

Екзаменаційна складова шкали дорівнює 40 % від R, а саме:

$$R_E = R_C \frac{0,4}{1 - 0,4} = 40 \text{ балів}$$

Таким чином, рейтингова шкала з дисципліни складає **R = R<sub>C</sub> + R<sub>E</sub> = 100 балів**.

Необхідною умовою допуску до екзамену є зарахування всіх лабораторних робіт та виконання МКР та курсової роботи, а також стартовий рейтинг студента (**R<sub>C</sub>**) не менше 40 % від **R<sub>C</sub>**, тобто 24 бали.

Виходячи з розміру шкали **R<sub>E</sub> = 40 балів**, критерії екзаменаційного оцінювання з використанням традиційних та ECTS оцінок наведені в таблиці:

<b>RD = R<sub>C</sub> + R<sub>E</sub></b>	<b>Оцінка ECTS</b>	<b>Традиційна оцінка</b>
95..100	A	відмінно
85..94	B	добре
75...84	C	
65...74	D	задовільно
60...64	E	
<b>RD ≤ 59</b>	Fx	незадовільно
<b>R<sub>C</sub> &lt; 24</b> або не виконані інші умови допуску до екзамену	F	недопущений

Склав: \_\_\_\_\_ ас. кафедри СП Булах Б.В.

(посада викладача, прізвище та ініціали, підпис)

Ухвалено на засіданні кафедри  
Протокол № \_\_\_\_\_ від \_\_\_\_\_ 2014

Завідувач кафедри \_\_\_\_\_ Петренко А.І.