

ЗАТВЕРДЖЕНО
Наказ ректора Київського національного
університету імені Тараса Шевченка
від «__» _____ 2017 року за № ____
згідно Ухвали Вченої Ради від 30.06.2017 р.

Додаток 5

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

(назва факультету, інституту, центру, коледжу)

Кафедра (циклова комісія)
(для коледжів)

«ЗАТВЕРДЖУЮ»
Заступник декана/директора
з навчальної роботи

«__» _____ 20__ року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування

(повна назва навчальної дисципліни)

для студентів

галузь знань	12 Інформаційні технології (шифр і назва)
спеціальність	124 Системний аналіз (шифр і назва спеціальності)
освітній рівень	Б (молодший бакалавр, бакалавр, магістр)
освітня програма	Системний Аналіз (назва освітньої програми)
спеціалізація (за наявності)	бакалавр (молодший бакалавр, бакалавр, магістр)
вид дисципліни	за вибором

Форма навчання	Денна
Навчальний рік	2017/2018
Семестр	3-4
Кількість кредитів ECTS	7
Мова викладання, навчання та оцінювання	Українська
Форма заключного контролю	Залік, Екзамен

Викладачі: ас. Денисов С.В., ас. Оноцький В.В.

Пролонговано: на 20__/20__ н.р. _____ (____)» _____ » 20__ р.

(підпис ,ПІБ, дата)

20__/20__ н.р. _____ (____)» _____ » 20__ р.

(підпис ,ПІБ, дата)

Розробники:

Клюшин Дмитро Анатолійович, д.ф.м.н, професор кафедри обчислювальної математики,
Денисов Сергій Вікторович, асистент кафедри обчислювальної математики

«ЗАТВЕРДЖЕНО»

Зав. кафедри _____

_____ (_____)
(підпис) (прізвище та ініціали)

Протокол № ____ від «__» _____ 20__р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2017 року № ____

Голова науково-методичної комісії _____ (_____)
(підпис) (прізвище та ініціали)

1. Мета дисципліни

Оволодіння компетенцією ефективного використання ООП для створення алгоритмічно та технологічно навантажених програмних проектів. Знайомство з підходами до розробки індустріального та академічного програмного забезпечення.

2. Попередні вимоги до опанування або вибору навчальної дисципліни (за наявності)¹:

1. Знати основні поняття програмування, мову C/C++ на базовому рівні, алгебру та математичний аналіз на базовому рівні, основні структури даних та алгоритми їх обробки
2. Вміти створювати програми на мові C/C++, читати та аналізувати математичні тексти
3. Володіти елементарними навичками пошуку та аналізу інформації

3. Анотація навчальної дисципліни:

Об'єктно-орієнтоване програмування є одним з основних підходів до розробки великих програмних проектів. Глибоке розуміння його концепцій, переваг та недоліків, разом з практичними навичками створення програм в рамках цієї парадигми на кількох мовах програмування є обов'язковою компетенцією сучасного ІТ спеціаліста. В рамках курсу студенти матимуть змогу практикувати ООП в мовах C++ та C#.

4. Завдання (навчальні цілі):

Закласти основи фундаментальної фахової підготовки, яка очікується від випускників класичних університетів із спеціальності "Системний аналіз"

5. Результати навчання за дисципліною:

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
1	Концепції та прийоми ООП	Лекції, майстер-класи	Модульні роботи	30%
2	Створювати складні програми мовами C++ та C#	лабораторні роботи, майстер-класи	Контрольні роботи, оцінки за лабораторні роботи	30%
3	Здатність працювати в команді	лабораторні роботи	Оцінки за лабораторні роботи	10%
4	Здатність самостійно знаходити інформацію та виконувати складні завдання	лабораторні роботи	Оцінки за лабораторні роботи	30%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання (необов'язково для вибіркових дисциплін які не входять до блоків спеціалізації)

Результати навчання дисципліни (код)	1.1	1.2	2.1	4.1	4.2
Програмні результати навчання (назва)					

¹альтернативний варіант:

Успішне опанування курсу _____

Знання теоретичних основ _____

7.Схема формування оцінки.

7.1Форми оцінювання студентів:

-семестрове оцінювання:

1. 3 лабораторні роботи по 10 балів
2. дві модульні контрольні роботи по 10 балів
3. поточний контроль - 10 балів

-підсумкове оцінювання (у формі заліку в першому семестрі, 40 балів, та екзамену в другому семестрі, 40 балів)¹:

-умови допуску до підсумкового екзамену:

Отримання не менше ніж 20 балів за результатами семестру.

7.2 Організація оцінювання: (обов'язково зазначається порядок організації передбачених робочою навчальною програмою форм оцінювання із зазначенням орієнтовного графіку оцінювання).

Перший семестр:

1. Лабораторна робота №1 – до 30 вересня
2. Модульна контрольна №1 – 15 жовтня
3. Лабораторна робота №2 – до 30 жовтня
4. Модульна контрольна №2 – 25 листопада
5. Лабораторна робота №3 – до 5 грудня

Другий семестр:

1. Лабораторна робота №1 – до 2 березня
2. Модульна контрольна №1 – 15 березня
3. Лабораторна робота №2 – до 25 березня
4. Модульна контрольна №2 – 25 квітня
5. Лабораторна робота №3 – до 30 травня

7.3 Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно /Satisfactory	60-74
Незадовільно / Fail	0-59
Зараховано / Passed	60-100
Не зараховано / Fail	0-59

¹ Залік виставляється за результатами роботи студента впродовж усього семестру і не передбачає додаткових заходів оцінювання для успішних студентів.

8. Структура навчальної дисципліни.

Тематичний план 17 занять першого семестру

№ п/п	Номер і назва теми*	Кількість годин		
		лекції	лабораторні	Самостійна робота
Мотиваційний вступ				
1	Мотивація створення ООП, альтернативні підходи, тонкощі використання. Реалізація інкапсуляції та принципу DRY з та без ООП в C++.	2		
Повторення та поглиблений аналіз засобів реалізації ООП в мові C++				
2	Клас та об'єкт в C++. Поля та методи класу. Варіанти синтаксису та структури опису класу.	2	2	4
3	Інкапсуляція через специфікатори доступу. Звернення до об'єкту через вказівник. Ключове слово this.	2	2	4
4	Конструктор класу. Стандартні списки ініціалізації. Додаткові можливості з C++11 – пряма ініціалізація, initializer_list та «ініціалізатори в фігурних дужках». Деструктори.	2	2	4
5	Конструктор копіювання. Конструктор переносу (C++11). Мета та особливості використання.	2	2	4
6	Статичні члени класу, константи. Нюанси використання. Модифікатор const. Тип enum з областю видимості (C++11).	2	2	4
7	Дружні функції, дружні класи. Вкладені класи. Переваги і недоліки.	2	2	4
8	Перезавантаження операторів. Варіанти, нюанси використання. Особливості для операторів =, (), [], <<, >>.	2	2	4
9	Успадкування як ключовий момент ООП. Сценарії використання. Недоліки та критика. Альтернативи. Повтор базового синтаксису успадкування в C++.	2	2	4
10	Порядок виклику конструкторів та деструкторів при успадкуванні. Делегування та успадкування конструкторів (C++11). Зміни рівня доступу до членів класу при різних сценаріях успадкування.	2	2	4
11	Множинне успадкування. Повтор поняття поліморфізму, переваги та недоліки, альтернативи. Віртуальні функції як основа поліморфізму в C++.	2	2	4
12	Дослідження реалізації поліморфізму в C++.	2	2	4

	Пропуски в ієрархії. Абстрактні класи. Концепція інтерфейсу в ООП. Віртуальні деструктори.			
13	Обробка виключень. Виключення-об'єкти, варіанти перехоплення, використання поліморфізму. Обмеження виключень (C++11).	2	2	4
Основи узагальненого програмування				
14	Поняття шаблону в C++ (template). Мета, переваги та недоліки, приклади.	2	2	4
15	Шаблонні функції та шаблонні класи.	2	2	4
16	Елементи метапрограмування	2		2
Висновки та підсумки				
17	C++ як потужний класичний інструмент розробника. Місце C++ та ООП в сучасному світі створення програмного забезпечення. Перспективи розвитку.	2		
	Всього ³	34	28	58

*Примітка: слід зазначити також теми, винесені на самостійне вивчення

Загальний обсяг (перший семестр) 120 год.⁴, в тому числі (вибрати необхідне):

Лекцій – 34 год.

Лабораторні заняття – 28 год.

Самостійна робота - 58 год.

Тематичний план 10 занять другого семестру

№ п/п	Номер і назва теми*	Кількість годин		
		лекції	лабораторні	Самостійна робота
Мова С# та платформа .NET				
1	Знайомство с С#.NET – історія, базова архітектура, переваги та недоліки.	2		
2	Базовий синтаксис С# - вбудовані типи, змінні, керуючі конструкції.	2	2	4
3	Базовий синтаксис С# - масиви, методи, структури, типи-посилання та типи-значення	2	2	4
4	Основи реалізації ООП в С# - базовий синтаксис класів. Статичні члени класу.	2	2	4
5	Інкапсуляція, успадкування, поліморфізм в С#.	2	2	4
6	Інтерфейси як заміна множинному успадкуванню. Обробка виключень.	2	2	4
7	Узагальнення (generics). Колекції. Делегати та лямбда-вирази.	2	2	8
8	LINQ. Ідея, переваги та недоліки. Сценарії використання.	2	2	8
9	Автоматичне збирання сміття. Ідея, переваги та недоліки. Можливість впливу на garbage collector.	2	2	4
10	Розробка веб-додатків як одна з сфер застосування С#.Net. Підхід MVC, платформа ASP.NET MVC	2	4	10
	Всього ³	20	20	50

*Примітка: слід зазначити також теми, винесені на самостійне вивчення

Загальний обсяг 90 год.⁴, в тому числі (вибрати необхідне):

Лекцій – 20 год.

Лабораторні заняття – 20 год.

Самостійна робота - 50 год.

9. Рекомендовані джерела⁵:

Основна: (Базова)

1. Шилдт Г., Полный справочник по С++, 4-е изд. – М.: Издательский дом “Вильямс”, 2006
2. Кнут Д. Искусство программирования. Т. 1-3, 3-е изд. — М.: Издательский дом “Вильямс”, 2000.
3. Вандервурд Д., Джосаттис Н. Шаблоны С++: справочник разработчика. – М.: Издательский дом “Вильямс”, 2003.
4. Ключин Д.А. Полный курс С++. Профессиональная работа. – М.: Издательский дом “Вильямс”, 2004.
5. Троелсен Э., Язык программирования С# 5.0 и платформа .NET 4.5, 6-е изд. – М.: Издательский дом “Вильямс”, 2013
6. Шилдт Г., Полное руководство С# 4.0, – М.: Издательский дом “Вильямс”, 2011
7. <https://docs.microsoft.com/en-us/dotnet/csharp/index>

Додаткова:

1. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. – Спб.: Питер, 2004.
2. Александреску А. Современное проектирование на С++. – Издательский дом “Вильямс”, 2002
3. Буч Г., Объектно-ориентированный анализ и проектирование с примерами приложений на С++, 2-е изд., пер. с англ. М., "Издательство Бином", Спб.: "Невский диалект", 1999.
4. https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=MAuqZiG1_4404984382
5. https://mva.microsoft.com/ru/training-courses/-c-8622?l=DQMesDF1_7804984382
6. <https://codeeasy.net>

Як правило – до 20 джерел

10. Додаткові ресурси (за наявності):

³ У робочій програмі навчальної дисципліни для лекційних, семінарських, практичних і лабораторних занять зазначається (реальна) кількість годин (кратне 2 год. – час тивалості пари).

⁴ Загальна кількість годин, відведених на дану дисципліну згідно навчального плану.

⁵ В тому числі Інтернет ресурси