

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА

МЕХАНІКО-МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ  
Кафедра математичної фізики



«ЗАТВЕРДЖУЮ»  
Заступник декана  
з навчальної роботи

Харитонов О.М.

« 31 вересня » 2020 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування  
для студентів

галузь знань	11 «Математика та статистика»
спеціальність	111 «Математика»
освітній рівень	перший (бакалавр)
освітня програма	«Комп'ютерна математика»
вид дисципліни	обов'язкова

Форма навчання	денна
Навчальний рік	2020/2021
Семестр	2
Кількість кредитів ECTS	6
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладач: Обвінцев Олександр Вальдемарович, канд. техн. наук, доцент, доцент кафедри математичної фізики

Пролонговано: на 20<sup>21</sup>/20<sup>22</sup> н.р. « 31 » 08 20<sup>21</sup> р.  
на 20 /20 н.р. ( ) « » 20 р.

КИЇВ – 2020

Розробник<sup>1</sup>: Обвінцев Олександр Вальдемарович, кандидат технічних наук, доцент, доцент кафедри математичної фізики.

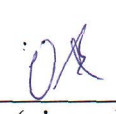
ЗАТВЕДЖЕНО  
Зав. кафедри  
Математичної фізики

 Самойленко В.Г.

Протокол № 1 від «28» 08 2020р.

Схвалено науково-методичною комісією механіко-математичного факультету

Протокол № 1 від «31» 08 2020р.

Голова науково-методичної комісії  професор, д.ф.-м.н. Олійник А.С.  
(підпис)

<sup>1</sup> Розробляється лектором. Робоча програма навчальної дисципліни розглядається на засіданні кафедри, науково-методичної комісії факультету/інституту, підписується завідувачем кафедри, головою науково-методичної комісії факультету/інституту і затверджується заступником декана/директора інституту з навчальної роботи.

1. **Мета дисципліни (до 300 символів)** – ознайомлення та оволодіння сучасними методами та теоретичними положеннями, притаманними інформатиці та програмній інженерії, та їх застосування при побудові алгоритмів та програм у парадигмі об'єктно-орієнтованого програмування для сучасних комп'ютерів.

2. **Попередні вимоги до опанування або вибору навчальної дисципліни:**

1. *Знати:* основні поняття програмування: виконавець, алгоритм, програма; синтаксис, семантику та властивості структур керування (ланцюг, розгалуження, цикли); методи побудови алгоритмів та програм, будову простих та складених типів даних; опис та використання підпрограм та модулів, синтаксичні конструкції мови програмування Python.

2. *Вміти:* будувати лінійні алгоритми та програми, розгалужені алгоритми та програми; будувати циклічні алгоритми та програми, програми, які оперують даними простих типів; будувати програми, які оперують даними складених типів: рядок, список, кортеж, словник, програми, що містять підпрограми, програми, що складаються з модулів.

3. **Анотація навчальної дисципліни (до 700 символів):**

Навчальна дисципліна «Об'єктно-орієнтоване програмування» є складовою освітньої програми підготовки фахівців за освітнім рівнем «бакалавр» галузі знань 11 математика та статистика зі спеціальності 111 математика освітньої програми «Комп'ютерна математика». У програмі дисципліни розглядаються такі фундаментальні поняття як файли, класи та об'єкти, наслідування та інкапсуляція, агрегація, виключення, ітератори та генератори, множинне наслідування, абстрактні класи та метакласи. Ці поняття необхідні для підготовки студентів до використання загальних методів програмування в подальших навчальних курсах, сприяння розвитку логічного та аналітичного мислення студентів.

**Дана дисципліна є обов'язковою.**

Викладається у 1 семестрі 1 курсу в обсязі **180 год.** (**6 кредитів ECTS<sup>2</sup>**) зокрема: *лекції – всього 28 год., лабораторні – 60 год. консультацій – 2 год, самостійної роботи студентів – 90 год.* У курсі передбачено **2 змістових модулі, 2 модульні контрольні роботи.** Завершується дисципліна **іспитом.**

4. **Завдання (навчальні цілі):**

формування здатності розв'язувати складні задачі та практичні проблеми у математиці або у процесі навчання, що передбачає застосування теорій та методів математики, статистики й комп'ютерних технологій і характеризується комплексністю та невизначеністю умов; набуття знань, умінь та навичок (компетентностей) на рівні новітніх досягнень у математиці, відповідно до освітнього рівня «Бакалавр». Зокрема, професійне оволодіння компетентностями:

- 1) Здатність до абстрактного мислення, аналізу та синтезу;
- 2) Здатність застосовувати знання у практичних ситуаціях;
- 3) Знання й розуміння предметної області та професійної діяльності;
- 4) Навички використання інформаційних і комунікаційних технологій;
- 5) Здатність учитися і оволодівати сучасними знаннями;
- 6) Здатність до пошуку, обробки та аналізу інформації з різних джерел;

- 7) Здатність приймати обґрунтовані рішення;
- 8) Здатність працювати автономно;
- 9) Здатність використовувати стандартні прийоми та методи математичних досліджень, проявляти творчий підхід, ініціативу;
- 10) Здатність формулювати проблеми математично та в символічній формі з метою спрощення їхнього аналізу й розв'язання;
- 11) Здатність здійснювати міркування та виокремлювати ланцюжки міркувань у математичних доведеннях на базі аксіоматичного підходу, а також розташовувати їх у логічну послідовність, у тому числі відрізнити основні ідеї від деталей і технічних викладок;
- 12) Здатність до кількісного мислення;
- 13) Здатність застосовувати чисельні методи для дослідження математичних моделей;
- 14) Здатність до аналізу математичних структур, у тому числі до оцінювання обґрунтованості й ефективності використовуваних математичних підходів;
- 15) Здатність застосовувати спеціалізовані мови програмування та пакети прикладних програм;
- 16) Здатність використовувати обчислювальні інструменти для чисельних і символічних розрахунків;
- 17) Здатність розуміти проблеми та виділяти їхні суттєві риси;
- 18) Здатність отримувати якісну інформацію на основі кількісних даних;
- 19) Здатність розробляти експериментальні та спостережні дослідження й аналізувати дані, отримані на їх основі;
- 20) Здатність пояснювати математичними термінами результати, отримані під час розрахунків;

##### 5. Результати навчання за дисципліною:

Результат навчання (РН) (1. знати; 2. вміти; 3. комунікація)		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 (Формуються розробником)			
РН 1.1	Знати поняття класу та об'єкту, наслідування та інкапсуляції	<i>Лекція, лабораторні заняття</i>	<i>Модульна контрольна робота 1, екзамен, активна робота на лекції, усні відповіді</i>	5%
РН 1.2	Знати поняття помилки та виключної ситуації, правила обробки помилок			10%
РН 1.3	Знати поняття ітератора та генератора; опис ітераторів та генераторів у мові Python; поняття декоратора функції		<i>Модульна контрольна робота 2, екзамен, активна робота на лекції, усні відповіді</i>	10%
РН 1.4	Знати поняття множинного наслідування; поняття метакласів, абстрактних класів, метапрограмування			10%
РН 2.1	Вміти будувати програми, що містять класи та об'єкти; будувати програми з	<i>Лекція, лабораторні</i>	<i>Модульна контрольна</i>	20%

	обробкою помилок та виключних ситуацій		робота 1, екзамен, виконання завдань, винесених на самостійну роботу	
РН 2.2	Вміти застосовувати у програмах ітератори та генератори; будувати програми, які використовують декоратори	<i>е заняття, самостійна робота</i>		20%
РН 2.3	Вміти будувати програми, які містять класи, що успадковують від декількох класів, абстрактні класи, метакласи; програми з використанням графічного інтерфейсу	<i>Лекція, лабораторн е заняття, самостійна робота</i>	<i>Модульна контрольна робота 2, екзамен, виконання завдань, винесених на самостійну роботу</i>	20%
РН 3.1	Здатність обґрунтовувати власний погляд на задачу та формулювати робочі гіпотези, спілкуватися з колегами з питань застосування методів та теорій		<i>Активна робота на лекції, лабораторних заняттях, усні відповіді</i>	2,5%
РН 3.2.	Вироблення навиків командної роботи	<i>Лекція, лабораторн е заняття</i>		2,5%

## 6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни	Р Н	Р Н	Р Н	Р Н	Р Н	Р Н	Р Н	Р Н	Р Н	Р Н
<b>Програмні результати навчання</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>3</b>	<b>3</b>	
	·	·	·	·	·	·	·	·	·	·
	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	
<b>РН-5</b> - Мати навички використання спеціалізованих програмних засобів комп'ютерної та прикладної математики і використовувати інтернет-ресурси;	+	+	+	+	+	+	+	+	+	+
<b>РН-21</b> - Розв'язувати типові задачі математичного аналізу, алгебри, диференціальних та інтегральних рівнянь, оптимізації за допомогою чисельних методів;	+	+	+	+	+	+	+	+	+	+
<b>РН-26</b> - Бути наполегливим у досягненні мети під час вирішення математичної проблеми					+	+	+	+	+	+

## 7. Схема формування оцінки:

### 7.1. Форми оцінювання студентів:

**- оцінювання впродовж навчального періоду:**

1. Командна робота студентів: РН1.1, РН1.2, РН1.3, РН1.4, РН3.1, РН3.2 – 10 балів/6 балів;
  2. Виконання завдань, винесених на самостійну роботу: РН2.1, РН2.2, РН2.3 – 20 балів/12 балів;
  3. Модульна контрольна робота 1: РН1.1, РН1.2, РН2.1 – 10 балів/6 балів;
  4. Модульна контрольна робота 2: РН1.3, РН1.4, РН2.2, РН2.3 – 10 балів/6 балів;
  6. Розв'язання задач на лабораторних заняттях: РН2.1, РН2.2, РН2.3 – 10 балів/5 балів;
- Разом 60/35

**- підсумкове оцінювання: іспит.**

- максимальна кількість балів, які можуть бути отримані: 40 балів;
- результати навчання, які будуть оцінюватись: РН1.1, РН1.2, РН1.3, РН1.4, РН2.1, РН2.2, РН2.3;
- форма проведення і види завдань: письмова робота, побудова та налагодження програм за комп'ютером.

**7.2. Організація оцінювання:**

Самостійна робота передбачає виконання зазначених керівником курсу завдань практичного характеру.

Командна робота полягає у виконанні завдання протягом заняття командою з 3 (2) студентів під контролем викладача.

Модульні контрольні роботи та колоквиум проводяться в час після занять в формі письмово-усній за практичними та теоретичними питаннями курсу, запропонованих керівником курсу.

Допускається оцінювання за допомогою технологій дистанційного навчання.

Критично-розрахунковий мінімум балів за навчання впродовж семестру становить **20** балів, рекомендований мінімум, розрахований з урахуванням специфіки дисципліни становить **35** балів. Студенти, які протягом семестру набрали сумарно меншу кількість балів ніж рекомендований мінімум **35** балів для підвищення балів отримують можливість скласти домашні завдання, які були задані протягом семестру. Студенти, які набрали впродовж семестру та за рахунок додаткових етапів оцінювання сумарно меншу кількість балів ніж критично-розрахунковий мінімум **20** балів, до складання іспиту не допускаються.

У випадку відсутності студента з поважних причин відпрацювання та перездачі форм контролю здійснюються у відповідності до „Положення про організацію освітнього процесу в Київському національному університеті імені Тараса Шевченка” (2018), <http://www.univ.kiev.ua/pdfs/official/Organization-of-the-educational-process.pdf>.

Якщо студент набрав протягом семестру більше, ніж 50 балів, він може отримати додаткові бали у рейтинг без складання іспиту в кількості, що залежить від набраних балів у семестрі:

від 51 до 55 балів – додається 24 балів

від 56 до 60 балів – додається 34 бали

Набравши протягом семестру більше, ніж 50 балів, студент може підвищити свій рейтинг в разі успішної здачі іспиту.

Форма іспиту – письмово-усна. Екзаменаційний білет іспиту складається із 4 завдань, перші два з яких є теоретичними, два інших – задачі. Кожне завдання оцінюється від 0 до 10 балів. Всього за іспит можна отримати від 0 до 40 балів. Мінімальна кількість балів, які додаються до семестрових – 24 бали.

**Терміни проведення форм оцінювання:**

1. Модульна контрольна робота 1: на 5-му тижні навчального періоду.
2. Модульна контрольна робота 2: на 10-му тижні навчального періоду.

**7.3. Шкала відповідності оцінок**

<b>Відмінно / Excellent</b>	90-100
<b>Добре / Good</b>	75-89
<b>Задовільно / Satisfactory</b>	60-74
<b>Незадовільно / Fail</b>	0-59

**8. Структура навчальної дисципліни. Тематичний план лекцій і лабораторних занять**

**2-й семестр**

Теми	Назва теми	Кількість годин				
		Лекції	Лабораторні заняття	Самост. робота	Контр. модульна робота	Інші форми контролю
<b>Змістовий модуль 1. Класи та об'єкти</b>						
1	Файли	2	4	6		
2	Класи та об'єкти	2	4	8		
3	Наслідування та агрегування	4	8	10		
4	Обробка помилок та виключних ситуацій	2	6	8		
5	Ітератори та генератори	2	4	10	2	
<b>Змістовий модуль 2. Специфічне використання класів</b>						
1	Декоратори	2	4	8		
2	Множинне наслідування	2	4	8		
3	Метакласи та метапрограмування	4	10	12		
4	Графічний інтерфейс	8	16	20		
Всього годин за 2 семестр		28	60	90		

**Загальний обсяг 180 годин, у тому числі:**  
**лекції – 28 годин,**  
**лабораторні заняття – 60 годин,**  
**консультації – 2 години,**  
**самостійна робота – 90 годин.**

**9. Рекомендовані джерела:**

**Основна література:**

1. Обвінцев О.В. Інформатика та програмування. Курс на основі Python. Матеріали лекцій. – К., Основа, 2017
2. Обвінцев О.В. Об'єктно-орієнтоване програмування. Курс на основі Python. Матеріали лекцій. – К., Основа, 2017
3. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar), <http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
4. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
5. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р., <http://www.matfiz.univ.kiev.ua/books>

6. Bernd Klein. PythonCourse <http://www.python-course.eu/index.php>

**Додаткова література:**

1. Самоучитель Python. <http://pythonworld.ru/samouchitel-python>
2. С. Шапошникова. Основы программирования на Python. Версия 2 (2011). <http://younglinux.info/pdf>
3. Бизли Д. Python. Подробный справочник. - Символ-Плюс, 2010.
4. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
5. Doug Hellmann. Python Module of the Week. - <http://pymotw.com/2/#>
6. Matt Telles. Python Power! The Comprehensive Guide. - Thomson Course Technology, 2012.
7. MarkLutz. Programming Python 4th Edition. - O'Reilly Media, 2010.
8. Tarek Ziadé. Expert Python Programming. - Packt Publishing, 2008.