

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ПРИКАРПАТСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТЕФАНИКА



Факультет математики та інформатики
Кафедра диференціальних рівнянь і прикладної математики

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

Рівень вищої освіти: Перший (бакалаврський)

Освітня програма: Комп'ютерне моделювання та
технології програмування

Спеціальність: 113 Прикладна математика

Галузь знань: 11 Математика та статистика

Затверджено на засіданні кафедри
Протокол №1 від 31 серпня 2023 р.

ЗМІСТ

1. Загальна інформація
2. Анотація навчальної дисципліни
3. Мета та цілі навчальної дисципліни
4. Програмні компетентності
5. Програмні результати навчання
6. Структура навчальної дисципліни
7. Система оцінювання навчальної дисципліни
8. Ресурсне забезпечення навчальної дисципліни
9. Політика навчальної дисципліни

1. ЗАГАЛЬНА ІНФОРМАЦІЯ

Назва дисципліни	Об'єктно-орієнтоване програмування Object-Oriented Programming
Освітня програма	Комп'ютерне моделювання та технології програмування
Спеціальність	113 Прикладна математика
Галузь знань	11 Математика та статистика
Освітній рівень	перший (бакалаврський)
Статус дисципліни	нормативна (з циклу професійної підготовки)
Рік підготовки/семестр	2 ^{ий} / 3 ^{ий}
Обсяг дисципліни	6 кредитів ECTS / 180 год.
Розподіл годин за видами занять	лекції – 30 год лабораторні – 44 год самостійна робота – 106 год
Форма контролю	екзамен
Мова викладання	українська
Посилання на сайт дистанційного навчання	https://d-learn.pnu.edu.ua/oop
Пререквізити	дискретна математика, програмування на Python/C++
Постреквізити	курсний проєкт з програмування, спеціалізовані мови програмування, практикум з моделювання і програмування, виробнича практика

Кафедра	диференціальних рівнянь і прикладної математики ауд. 315 ЦК, https://kdrpm.pnu.edu.ua
Викладач(-и)	Віктор МАЗУРЕНКО
Контактний телефон	(0342)596027
E-mail	viktor.mazurenko@pnu.edu.ua
Профайл	https://mazurenko.pnu.edu.ua
Консультації	згідно з розкладом консультацій на сайті кафедри

2. АНОТАЦІЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

У цьому курсі студенти спеціальності «Прикладна математика» мають нагоду заглибитись у світ об'єктно-орієнтованого програмування (де будь-яку програму розглядають як сукупність «об'єктів», які взаємодіють між собою і кожен з яких є екземпляром якогось «класу»), освоїти його ключові концепції та практики. Така парадигма програмування є хоч і не надто простою, але вельми ефективною при розробці складних програмних проєктів. Завдяки таким ключовим концепціям як «абстрагування», «інкапсуляція», «успадкування» і «поліморфізм», програми, написані в об'єктно-орієнтованому стилі, володіють більшою модульністю і гнучкістю, читабельністю і оптимізованістю коду, захищеністю даних і адаптованістю до змін, ніж програми, написані в інших стилях. Розуміння концепцій об'єктно-орієнтованого програмування, його переваг і недоліків разом з практичними навиками написання програм мовою Python в рамках цієї парадигми дозволить студентам бути успішним в області сучасних технологій програмування.

3. МЕТА І ЦІЛІ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Набуття студентами фахових компетентностей з об'єктно-орієнтованого програмування для створення програмного забезпечення, яке є більш розширюваним, підтримуваним і зрозумілим як для розробників, так і користувачів. Ознайомити студентів з основними поняттями і концепціями об'єктно-орієнтованого програмування. Сформулювати вміння писати програми мовою Python в об'єктно-орієнтованому стилі для автоматизації задач різної складності. Сприяти розвитку навиків командної роботи та спільної програмної розробки.

4. ПРОГРАМНІ КОМПЕТЕНТНОСТІ

Відповідно до освітньо-професійної програми «Комп'ютерне моделювання та технології програмування» для першого (бакалаврського) рівня вищої освіти:

- ФК04.** Здатність розробляти алгоритми та структури даних, програмні засоби та програмну документацію.
- ФК05.** Здатність проектувати бази даних, інформаційні системи та ресурси.
- ФК07.** Здатність експлуатувати та обслуговувати програмне забезпечення автоматизованих та інформаційних систем різного призначення.
- ФК08.** Здатність використовувати сучасні технології програмування та тестування програмного забезпечення.
- ФК09.** Здатність до проведення математичного і комп'ютерного моделювання, аналізу та обробки даних, обчислювального експерименту, розв'язання формалізованих задач з допомогою спеціалізованих програмних засобів.

5. ПРОГРАМНІ РЕЗУЛЬТАТИ НАВЧАННЯ

- PH11.** Вміти застосовувати сучасні технології програмування та розроблення програмного забезпечення, програмної реалізації чисельних і символьних алгоритмів.
- PH13.** Використовувати в практичній роботі спеціалізовані програмні продукти та програмні системи комп'ютерної математики.
- PH14.** Виявляти здатність до самонавчання та продовження професійного розвитку.

6. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ТЕМА, ПЛАН	КІЛЬКІСТЬ ГОДИН		
	ЛЕКЦІЇ	ЛАБОРАТОРНІ	САМОСТІЙНА РОБОТА
ФУНКЦІЇ - поняття підпрограми і функції, визначення і виклик функцій, повернення кількох значень - аргументи (параметри) функції: формальні і фактичні, передача аргументів: позиційні та іменовані (ключові) аргументи; типові (за замовчуванням) значення аргументів - функції з довільною кількістю аргументів, області видимості: локальні і глобальні змінні - поняття про функційний тип даних, анонімні (lambda-функції) - рекурсія і рекурсивні функції, генератор-функції - декоратори для функцій	4	6	10
КЛАСИ І ОБ'ЄКТИ, АТРИБУТИ І МЕТОДИ - основні парадигми програмування та об'єктно-орієнтований стиль програмування - класи і об'єкти (екземпляри) класу, атрибути і методи класу/об'єкта, абстрагування і абстракція - створення класу, посилання self на об'єкт класу, конструктор __init__ (ініціалізація) і деструктор __del__ (знищення) об'єкта - створення і знищення об'єктів - звернення до атрибутів і методів об'єкта, атрибути об'єкта і атрибути класу, додавання і вилучення атрибутів і методів - методи і функції	4	6	10
КОНЦЕПЦІЯ ІНКАПСУЛЯЦІЇ - інкапсуляція - статичні атрибути і методи - вступ до UML: діаграма класів, видимість атрибутів і методів, відношення між класами	4	6	10

КОНЦЕПЦІЯ УСПАДКУВАННЯ - успадкування (наслідування) - перевизначення атрибутів і методів - особливості роботи з методом-конструктором, виклик методів базового класу, суперклас object - успадкування на діаграмах класів - множинне і багатократне успадкування - проблеми, пов'язані з множинним успадкуванням, Method Resolution Order - виклик методів базових класів	6	6	10
КОНЦЕПЦІЯ ПОЛІМОРФІЗМУ - поліморфізм - віртуальні методи	2	6	10
СПЕЦІАЛЬНІ (МАГІЧНІ) АТРИБУТИ І МЕТОДИ - спеціальні атрибути - спеціальні методи - спеціальні методи для колекцій - перевантаження операторів (арифметичні, присвоєння, порівняння, унарні) - ітератори і генератори	4	6	10
КЛАСИ ВИКЛЮЧЕНЬ І АБСТРАКТНІ КЛАСИ - класи виключень - абстрактні класи - інтерфейси та шаблони програмування - домішки	4	6	10
МОДУЛЬНИЙ КОНТРОЛЬ	–	2	18
ПІДСУМКОВИЙ КОНТРОЛЬ	–	–	18
ВСЬОГО	30	44	106

7. СИСТЕМА ОЦІНЮВАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Накопичування балів впродовж вивчення дисципліни

Види навчальної роботи	Максимальна кількість балів
Лекції (ЛК), самостійна робота (СР)	5
Лабораторні роботи (ЛР)	35
Модульний контроль (МК)	10
Підсумковий контроль (ПК)	50
Максимальна кількість балів	100

Види навчальної роботи	Навчальні тижні																	Разом	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	КС		С
ЛК, СР																	5		5
ЛР			5		5		5		5		5		5		5				35
МК															10				10
ПК																		50	50
Разом			5		5		5		5		5		5		15		5	50	100

ШКАЛА ОЦІНЮВАННЯ: НАЦІОНАЛЬНА ТА ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проєкту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
80 – 89	B	добре	
70 – 79	C		
60 – 69	D	задовільно	
50 – 59	E		
26 – 49	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-25	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

КРИТЕРІЇ ОЦІНЮВАННЯ НАВЧАЛЬНИХ ДОСЯГНЕНЬ

(відповідно до Положення про організацію освітнього процесу)

– **«відмінно»** – здобувач освіти міцно засвоїв теоретичний матеріал, глибоко і всебічно знає зміст навчальної дисципліни, основні положення рекомендованої літератури, логічно мислить і будує відповідь, вільно використовує набуті теоретичні знання при аналізі практичного матеріалу, висловлює своє ставлення до тих чи інших проблем, демонструє високий рівень засвоєння практичних навичок;

– **«добре»** – здобувач освіти добре засвоїв теоретичний матеріал, володіє основними аспектами рекомендованої літератури, аргументовано викладає його; має практичні навички, висловлює свої міркування з приводу тих чи інших проблем, але припускається певних неточностей і похибок у логіці викладу теоретичного матеріалу або при аналізі практичного матеріалу;

– **«задовільно»** – здобувач освіти в основному опанував теоретичними знаннями навчальної дисципліни, орієнтується в першоджерелах та рекомендованій літературі, але непереконливо відповідає, плутає поняття, додаткові питання викликають невпевненість або відсутність стабільних знань; відповідаючи на запитання практичного характеру, виявляє неточності у знаннях, не вміє оцінювати факти та явища, пов'язувати їх із майбутньою діяльністю;

– **«незадовільно»** – здобувач освіти не опанував навчальний матеріал дисципліни, не знає наукових фактів, визначень, майже не орієнтується в рекомендованій літературі, відсутнє наукове мислення, практичні навички не сформовані.

8. РЕСУРСНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Матеріально-технічне забезпечення	Лекційна аудиторія, мультимедіа, комп'ютерна лабораторія з програмним забезпеченням
Навчально-методичне забезпечення	Навчальний контент на освітніх платформах https://d-learn.pnu.edu.ua/oop https://classroom.google.com/oop
Інформаційне забезпечення	Рекомендована література

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Крєневич А.П. Python у прикладах і задачах. Ч. 2. Об'єктно-орієнтоване програмування: навч. посібн. – К.: ВПЦ "КУ", 2020. – 152 с. – Режим доступу: [url](#)
2. Мізюк О. Путівник мовою програмування Python [Електронний ресурс] – 2019. – Режим доступу: [url](#)
3. Мізюк О. Практикум з програмування мовою Python [Електронний ресурс] – 2019. – Режим доступу: [url](#)
4. Татарчук Д.Д., Діденко Ю.В. Об'єктно-орієнтоване програмування мовою PYTHON : Конспект лекцій [Електронний ресурс]. – К.: КПІ ім. Ігоря Сікорського, 2021. – 129 с. – Режим доступу: [url](#)
5. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns. Elements of Reusable Object-Oriented Software. – Addison-Wesley Professional, 1994. – Access mode: [url](#)
6. Swaroop Ch. A byte of Python. – Пер. В. Смоляр. – 2013. – 151 с. – Режим доступу: [url](#)
7. The Python Tutorial [Електронний ресурс] – Режим доступу: [url](#)

9. ПОЛІТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Академічна доброчесність	Важливим є надання достовірної інформації про результати власної навчальної (наукової, творчої) діяльності та посилання на джерела інформації у разі використання ідей, відомостей, розробок. Плагіат та інші види академічної недоброчесності не принесуть користі, тому є недоречними та контролюються
Пропуски занять (відпрацювання)	Наслідком періодичних пропусків є самостійне опрацювання навчального матеріалу з можливим консультуванням у викладача відповідно до графіка консультацій. Наслідком систематичних пропусків є додаткові види самостійної роботи, які контролюються на передбачених у графіку навчального процесу тижнях контролю самостійної роботи

Виконання завдань пізніше встановленого терміну	Приводить до втрати частини балів, запланованих у системі оцінюванні навчальної дисципліни (усі види навчальної роботи важливо виконувати належним чином і вчасно, щоб зберегти загальний темп курсу, котрий сприяє ефективному засвоєнню матеріалу без шкоди здоров'ю)
Невідповідна поведінка під час заняття	Приводить до відсторонення від заняття
Додаткові бали	До 5 балів за активність, комунікативність, креативність, наполегливість, самостійність при вивченні дисципліни
Неформальна освіта	Можливість повного або часткового зарахування результатів неформальної освіти за умови співпадіння програмних результатів навчання

Викладач В.Мазуренко