

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ

ЗАТВЕРДЖУЮ
Декан математичного факультету



С.І. Гоменюк
(ініціали та прізвище)

«01» вересня 2025 р.

СИЛАБУС ОСВІТНЬОГО КОМПОНЕНТА
КУРСОВА РОБОТА З ДИСЦИПЛІНИ
«ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»

підготовки _____ бакалаврів _____

денної та заочної форм здобуття освіти

освітньо-професійна програма _____ Комп'ютерні науки _____

спеціальності _____ 122 - Комп'ютерні науки _____

галузі знань _____ 12 Інформаційні технології _____

ВИКЛАДАЧ: Миронова Н. О., к.т.н., доцент, доцент каф. комп'ютерних наук
Борю С. Ю., к.т.н., доцент, доцент каф. комп'ютерних наук
Решевська К. С., к.т.н., доцент, доцент каф. комп'ютерних наук

Обговорено та ухвалено
на засіданні кафедри комп'ютерних наук

Протокол №1 від «25» серпня 2025 р.
Завідувач кафедри комп'ютерних наук

_____ Г. М. Шило

Погоджено

Гарант освітньо-професійної програми

_____ Н. В. Матвіїшина

2025 рік



Зв'язок з викладачем: Миронова Наталія Олексіївна

E-mail: natali.myronova@gmail.com

Зв'язок з викладачем: Решевська Катерина Сергіївна

E-mail: reshka82zp@gmail.com

Зв'язок з викладачем: Борю Сергій Юрійович:

E-mail: bsu@znu.edu.ua

Сезн ЗНУ повідомлення: <https://moodle.znu.edu.ua/course/view.php?id=18038>

Телефон (кафедра): 289-12-57

Інші засоби зв'язку: Viber, Telegram

Кафедра комп'ютерних наук, ауд. №39, 1 корпус ЗНУ

1. Опис освітнього компонента

Курсова робота є видом індивідуального навчально-дослідного завдання в рамках освітнього компонента «Об'єктно-орієнтоване програмування» для здобувачів бакалаврського рівня спеціальності 122 Комп'ютерні науки, освітньо-професійної програми Комп'ютерні науки.

Метою написання курсової роботи є формування у здобувачів вищої освіти системного розуміння теоретичних основ і практичних навичок застосування об'єктно-орієнтованого підходу до розроблення програмного забезпечення. Завдання курсової роботи полягає в опануванні принципів інкапсуляції та організації тексту програми, використанні конструкторів і деструкторів для управління ресурсами, виконанні перевантаження операторів, а також застосуванні механізмів успадкування та поліморфізму для побудови гнучких і масштабованих програмних архітектур. Особлива увага приділяється розробці власних динамічних структур даних і роботі зі стандартною бібліотекою шаблонів (STL), впровадженню механізмів обробки винятків, а також реалізації багатопоточної та асинхронної взаємодії у програмних застосунках. Виконання курсової роботи сприяє розвитку компетентностей зі створення сучасних програмних рішень відповідно до вимог галузі комп'ютерних наук.

Пререквізити: підґрунтям для написання курсової роботи з дисципліни «Об'єктно-орієнтоване програмування» є ОК: «Об'єктно-орієнтоване програмування», «Організація та обробка електронної інформації».

Компетентності, отримані під час виконання курсової роботи з дисципліни «Об'єктно-орієнтоване програмування», використовуються для написання розділів кваліфікаційної роботи бакалавра.

Паспорт навчальної дисципліни

Нормативні показники	денна форма здобуття освіти	заочна форма здобуття освіти
1	2	3
Статус ОК	Обов'язкова	
Семестр	3-й	3-й
Кількість кредитів ECTS	1	1
Кількість годин	30	30
Консультації	За розкладом; дистанційно	
Вид підсумкового семестрового контролю:	захист курсової роботи	
Посилання на електронний контент освітнього компонента у СЕЗН ЗНУ (платформа Moodle)	https://moodle.znu.edu.ua/course/view.php?id=18038	

2. Методи досягнення запланованих освітньою програмою компетентностей і результатів навчання

Компетентності/ результати навчання	Методи навчання	Форми і методи оцінювання
Компетентності		
ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.	пояснення, обговорення, самостійна робота	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання
ЗК2. Здатність застосовувати знання у практичних ситуаціях.	пояснення, обговорення, проєктна робота, самостійна робота	Підсумковий контроль: пояснювальна записка, демонстрація функціональної бази даних або інформаційної системи, усний захист курсової роботи
ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.	пояснення, обговорення, проєктна робота, самостійна робота	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання
ЗК6. Здатність вчитися й оволодівати сучасними знаннями.	робота з науковою та професійною літературою, самостійне опрацювання матеріалу, онлайн-курси	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання

Компетентності/ результати навчання	Методи навчання	Форми і методи оцінювання
ЗК7 Здатність до пошуку, оброблення та аналізу інформації з різних джерел	пояснення, обговорення, самостійна робота	Підсумковий контроль: пояснювальна записка, демонстрація функціональної бази даних або інформаційної системи, усний захист курсової роботи
ЗК11 Здатність приймати обгрунтовані рішення	робота з науковою та професійною літературою, самостійне опрацювання матеріалу, онлайн-курси	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання
ЗК16 Здатність ухвалювати рішення та діяти, дотримуючись принципу неприпустимості корупції та будь-яких інших проявів недоброчесності	робота з науковою та професійною літературою, самостійне опрацювання матеріалу, онлайн-курси	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання
СК1 Здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обгрунтовування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування	пояснення, демонстрування, виконання завдань лабораторних та самостійних робіт	Підсумковий контроль: пояснювальна записка, демонстрація функціональної бази даних або інформаційної системи, усний захист курсової роботи
СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління	пояснення, демонстрування, самостійна робота, проектні роботи	Підсумковий контроль: пояснювальна записка, демонстрація функціональної бази даних або інформаційної системи, усний захист курсової роботи
Результати навчання		
ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.	пояснення, пояснення, демонстрування, самостійна робота, проектні роботи	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання

Компетентності/ результати навчання	Методи навчання	Форми і методи оцінювання
ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.	пояснення, демонстрування, самостійна робота, проєктні роботи	Підсумковий контроль: пояснювальна записка, демонстрація функціональної бази даних або інформаційної системи, усний захист курсової роботи
ПР20 Вміти донести до фахівців і нефахівців інформації, ідеї, проблеми, рішення, власний досвід та аргументацію; презентувати професійні погляди, ідеї та результати роботи в сфері комп'ютерних наук, забезпечуючи порозуміння та згоду всіх учасників комунікаційного процесу	пояснення, самостійна робота, проєктні роботи	Поточний контроль: консультації з керівником курсової роботи, перевірка окремих етапів виконання

3. Тематика курсових робіт

Здобувачам вищої освіти надається право вільного вибору теми згідно запропонованих тем курсових робіт. Здобувачі вищої освіти можуть пропонувати свої теми, які відповідають змісту освітнього компоненту «Об'єктно-орієнтоване програмування», і мають бути узгоджені з керівником курсової роботи або лектором освітнього компоненту.

Здобувачі вищої освіти зобов'язані обрати тему впродовж місяця з початку занять. Здобувачі вищої освіти, які не обрали у встановлені терміни тему курсової роботи, керівник курсової роботи або лектор освітнього компоненту мають право призначити тему з переліку орієнтовних тем курсових робіт на власний розсуд.

Керівників курсової роботи призначають з числа викладачів кафедри і закріплюють розпорядженням завідуючого кафедри. Керівництво курсовою роботою здійснюється професорами, доцентами, а також викладачами, які мають досвід науково-педагогічної і практичної роботи.

Процедура реалізації процесу виконання курсової роботи в умовах дистанційного та змішаного навчання здійснюється за допомогою створення відповідного курсу або розділу у відповідній освітній компоненті в СЕЗН ЗНУ Moodle, який містить відповідне інформаційне, навчально-методичне забезпечення, бланки документів, необхідні розділи, в які здобувачі вищої освіти завантажують рукопис курсової роботи, презентацію та ін.

Згідно з діючим «Положенням про запобігання та виявлення академічного плагіату в курсових, кваліфікаційних роботах здобувачів вищої освіти,



дисертаціях на здобуття ступенів доктора філософії та доктора наук у Запорізькому національному університету» (https://www.znu.edu.ua/pidrozdily/viddil_monitoryngu/polozhennya_2024.pdf) курсові роботи здобувачів вищої освіти підлягають обов'язковій процедурі перевірки на наявність плагіату (рівня запозичень).

Перевірка рівня запозичень у курсовій роботі здійснюється кафедрою не пізніше ніж за три дні до захисту роботи і виконується відповідальною особою, призначеною завідувачем кафедри для введення до електронної системи StrikePlagiarism текстів курсових робіт.

Орієнтовна тематика курсових робіт на 2025/2026 навчальний рік:

1. Масив з автоматичним розширенням (Dynamic Array / Vector). Аналог **std::vector**, шаблон класу **Vector<T>**, що сам розширюється при досягненні ліміту.
2. Файл, як масив з автоматичним розширенням (FileVector). Шаблон класу, який надає інтерфейс для роботи з типизованим файлом, як з масивом.
3. Універсальний стек (Stack). Шаблон класу **Stack<T>**, що реалізує стек з базовими операціями: **push, pop, top, isEmpty**.
4. Черга (Queue). Шаблон класу **Queue<T>** з методами **enqueue, dequeue, peek**.
5. Однозв'язний список (**Linked1List**). Шаблон класу **Linked1List<T>**.
6. Двозв'язний список (**Linked2List**). Шаблон класу **Linked2List<T>**.
7. Калькулятор (Calculator). Шаблон класу для базових арифметичних операцій з різними типами чисел. Конструктор класу отримує рядок з арифметичним виразом, наприклад $5.6 + 4 * (7 - 67.7) / 56$. Кількість відкритих дужок не обмежена. У випадку некоректно заданого арифметичного виразу повідомляє про помилку.
8. Універсальний словник (Map / Dictionary). Шаблон класу **Map<KeyType, ValueType>** на основі хеш-таблиці або бінарного дерева.
9. Матриця (Matrix). Шаблон класу для двовимірної матриці з базовими операціями (додавання, множення, транспонування, визначник).
10. Довгі цілі числа (BigInteger). Шаблон класу для роботи з довгими цілими числами. Передбачити операції додавання та віднімання. Примітка: Довге ціле число – це ціле число, значення якого не може бути записано ні в жодний стандартний тип даних. Клас повинен зберігати одне таке ціле число, виконувати на ним математичні дії такі як додавання та віднімання іншого довгого числа або короткого.
11. Сортиувальник (Sorter). Шаблон з реалізацією різних алгоритмів сортування (**Bubble, Quick, Merge** тощо). Шаблон сортування з параметром політики сортування (за не зростанням, не спаданням).
12. Символьний рядок (String). Аналог класу **string**. Повинен підтримувати операції пошуку, реверсу, видалення, вставки. Додавання та видалення повинно супроводжуватись динамічним перерозподілом пам'яті. Клас **string** стандартної



бібліотеки використовувати заборонено.

13. Контейнер Points. Кожен елемент такого контейнеру має такі властивості: координата x (ціле значення), координата y (ціле значення), віддаленність точки від початку системи координат, колір точки, її порядковий номер. Контейнер підтримує операції додавання, видалення, пошук за вказаним параметром, сортування по вказаному параметру, пошук ареалів заданого кольору (кількість та площу), визначення максимального прямокутника, в межах якого існують точки (його площу, та координати вершин такого прямокутника).

14. Розробити шаблон класу-контейнер Students<T>, який зберігає об'єкти довільного типу T. Контейнер повинен підтримувати такі операції: додавання, вставку, видалення (за індексом та/або ітератором), пошук (з поверненням індексу та/або ітератора), сортування, доступ до елементів (get / set). Структура типу T (наприклад, **Student**) визначається в основній програмі.

15. Множина (Set). Розробити шаблон класів для роботи з множинами. Передбачити методи додавання, видалення, пошуку. Навігація по контейнеру за допомогою ітераторів.

16. Комплексне число (Complex). Розробити шаблон класів для роботи з комплексними числами. Передбачити основні операції над комплексними числами (Арифметика комплексних чисел).

17. Календар. Розробити клас для роботи з календарними датами. Передбачити можливість отримання дати через задану кількість секунд, додавання/віднімання днів, перевірка на високосний рік, обчислення різниці між датами.

18. Клас “**Вектор у 2D/3D**”. Операції над векторами: довжина, скалярний добуток, векторний добуток, кут між векторами.

19. Клас “Поліном”. Реалізувати клас для роботи з многочленами (додавання, множення, похідна, значення в точці).

20. Клас “Інтеграл”. Реалізувати клас, який знаходить числове значення інтегралу заданої функції на заданому інтервалі.

21. Клас “Багаторозрядна арифметика”. Реалізувати клас для роботи з багаторозрядними числами. Передбачити основні арифметичні операції.

22. Клас “Геометричні фігури”. Абстрактний клас Shape, похідні: Circle, Rectangle, Triangle. Методи: обчислення площі, периметра, масштабування.

23. Клас “Файлова система” (імітація структури файлів). Створити класи File, Directory, FileSystem, де можна створювати, видаляти, шукати файли/папки.

24. Клас “Текстовий редактор”. Клас, який може зберігати текст, додавати/видаляти символи, рахувати слова, абзаци, символи.

25. Гра “Pac-man”. Створити програмне забезпечення — спрощену версію гри Pac-Man, реалізовану з використанням принципів об'єктно-орієнтованого програмування, з чітким поділом відповідальностей між класами, підтримкою базової ігрової логіки та взаємодії користувача з грою. Реалізувати класи, що відповідають за: головного героя (Pac-Man), привидів (вороги), карту/лабіринт, їжу (пункти), ігровий процес (логіку гри, рахунок, стан гри).

Усі класи персонажів є похідними від єдиного абстрактного класу.

4. Структура навчальної дисципліни

№ з/п	Вид навчальної діяльності (завдання)	Кількість годин о/д. ф.	Кінцевий термін виконання
1	2	3	4
1	Ознайомлення з темою, розробка плану роботи	2	до 24.09.2025
2	Аналіз предметної області та постановка завдання	2	до 26.09.2025
3	Аналіз засобів та технологій розробки	2	до 15.10.2025
4	Оформлення розділів 1,2 курсової роботи	4	до 20.10.2025
5	Розробка програми	10	до 31.10.2025
6	Оформлення розділу 3 курсової роботи, додатків	4	до 10.11.2025
7	Оформлення списку використаних джерел	2	до 20.11.2025
8	Перевірка та доопрацювання курсової роботи	2	до 02.12.2025
9	Захист курсової роботи	2	до 11.12.2025

Методичні рекомендації до курсової роботи розміщено СЕЗН ЗНУ Moodle на сторінці освітнього компонента «Об'єктно-орієнтоване програмування»

5. Система та критерії оцінювання курсової роботи

№ з/п	Вид навчальної діяльності	Критерії оцінювання	Усього балів
1	Відповідність теми змісту роботи та аналіз предметної області	Критерії оцінювання: повнота та обґрунтованість аналізу предметної області, виділення об'єктів та їх ролей – 2 бали; чіткість формулювання мети, завдань і актуальності роботи – 2 бали; відповідність назви змісту роботи, логічність та послідовність структури – 2 бали. Рівні оцінювання етапу: 5–6 балів – тема повністю обґрунтована, аналіз предметної області глибокий і логічний; 3–4 бали – тема розкрита частково, є неточності у визначенні об'єктів або завдань; 1–2 бали – поверхневий або формальний аналіз, нечітка постановка задачі.	6
2	Об'єктно-орієнтоване проектування (класи, зв'язки, UML)	Критерії оцінювання: коректність виділення класів, їх атрибутів і методів – 4 бали; правильність визначення зв'язків між класами	10

№ з/п	Вид навчальної діяльності	Критерії оцінювання	Усього балів
		(наслідування, агрегація, композиція) – 3 бали; наявність і коректність UML-діаграм (діаграма класів, за потреби – послідовностей) – 3 бали. Рівні оцінювання етапу: 8–10 балів – проектування повне, логічне, відповідає принципам ООП; 4–7 балів – структура в цілому коректна, але має спрощення або неточності; 1–3 бали – відсутня або помилкова модель класів.	
3	Реалізація класів і принципів ООП	Критерії оцінювання: використання принципів інкапсуляції, наслідування, поліморфізму, абстракції – 4 бали; коректна реалізація конструкторів (за замовчуванням, копіювання, переміщення) та деструктора – 3 бали; винесення реалізації методів за межі опису класу – 2 бали. Рівні оцінювання етапу: 8–9 балів – реалізація повністю відповідає вимогам ООП; 5–7 балів – основні принципи дотримані, але є недоліки; 1–4 бали – грубі порушення принципів ООП.	10
4	Реалізація класу-контейнера	Критерії оцінювання: динамічне зберігання елементів та автоматичне розширення контейнера – 3 бали; реалізація базових методів (додавання, видалення, пошук, модифікація, розмір) – 3 бали; перевантаження обов'язкових операторів (==, =, ++, --, []) – 3 бали. Рівні оцінювання етапу: 8–9 балів – контейнер реалізований повністю, коректно працює; 5–7 балів – функціональність реалізована частково; 1–4 бали – контейнер має суттєві помилки або неповний.	10
5	Реалізація ітераторів	Критерії оцінювання: наявність вкладеного класу-ітератора – 2 бали; реалізація методів begin() та end() – 2 бали; підтримка операцій ++, --, *, ==, != – 2 бали.	6

№ з/п	Вид навчальної діяльності	Критерії оцінювання	Усього балів
		Рівні оцінювання етапу: 5–6 балів – ітератори повністю функціональні; 3–4 бали – реалізовані з обмеженнями; 1–2 бали – відсутні або працюють некоректно.	
6	Основна програма та демонстрація роботи	Критерії оцінювання: наявність основної програми, що демонструє всі можливості класів – 3 бали; коректність сценаріїв використання – 2 бали; відсутність прив'язки методів до конкретних файлів чи потоків – 1 бал. Рівні оцінювання етапу: 5–6 балів – демонстрація повна і наочна; 3–4 бали – демонстрація часткова; 1–2 бали – мінімальна або формальна демонстрація.	6
7	Тестування та якість коду	Критерії оцінювання: наявність не менше трьох тестів з описом перевірюваних випадків – 2 бали; дотримання стандартів стилю коду та коментування методів – 2 бали; обробка помилок і коректна робота програми – 2 бали. Рівні оцінювання етапу: 5–6 балів – тести повні, код якісний і читабельний; 3–4 бали – тестування обмежене; 1–2 бали – тести відсутні або формальні.	6
8	Документація	Критерії оцінювання: – наявність пояснювальної записки – 2 бали; – повнота та логічність опису структури програми – 2 бали; – повнота опису реалізованих алгоритмів і методів – 2 бали. Рівні оцінювання етапу: – 5–6 балів – документація повна, структурована, містить чіткий опис програми та алгоритмів; – 3–4 бали – документація наявна, але містить неповний або поверхневий опис; – 1–2 бали – документація фрагментарна або відсутня.	6
Усього			60
Підсумковий контроль / Захист курсової роботи			

№ з/п	Вид навчальної діяльності	Критерії оцінювання	Усього балів
Залік	Уміння розкривати в усній доповіді зміст розглянутих у курсовій роботі положень		20
	Компетентність у відповідях на запитання під час захисту курсової роботи		20
Усього			40

Шкала оцінювання ЗНУ: національна та ECTS

За шкалою ECTS	За шкалою університету	За національною шкалою	
		Екзамен	Залік
A	90 – 100 (відмінно)	5 (відмінно)	Зараховано
B	85 – 89 (дуже добре)	4 (добре)	
C	75 – 84 (добре)		
D	70 – 74 (задовільно)	3 (задовільно)	
E	60 – 69 (достатньо)		
FX	35 – 59 (незадовільно – з можливістю повторного складання)	2 (незадовільно)	Не зараховано
F	1 – 34 (незадовільно – з обов’язковим повторним курсом)		

6. Основні навчальні ресурси

Рекомендована література

1. Головачова О. В. Програмування на C++ I частина (оновлена) : навч. посіб. Одеса, 2024. 136 с.
2. Грицюк Ю. І., Рак Т. Є. Об'єктно-орієнтоване програмування мовою C++ : навч. посіб. Львів : Вид-во ЛДУ БЖД, 2011..
3. Кухарська Н. П., Лагун А. Е. Практикум з об'єктно-орієнтованого програмування. Мова C++ : навч. посіб. Львів : ННІ ПП НУ "Львівська політехніка", 2022. 186 с.
4. Lippman S. B. C++ Primer [Text] / Stanley B. Lippman, Josée Lajoie, Barbara E. Moo. 5th ed. Addison-Wesley Professional, 2012. 976 p.
5. Stroustrup B. The C++ Programming Language [Text] / Bjarne Stroustrup. – 4th ed. Addison-Wesley Professional, 2013. 1376 p.
6. Williams A. C++ Concurrency in Action. 2nd ed. Manning Publications, 2019. 592 p.

Інформаційні ресурси

1. C++ Reference – найповніша документація та довідник мови C++. *cppreference.com*. URL: <https://en.cppreference.com/>



2. C++ Resources Network - тьюторіали, довідник та форум спільноти. *cplusplus.com*. URL: <https://cplusplus.com>
3. Learn C++ - безкоштовний онлайн-підручник від основ до просунутих тем. *LearnCpp.com*. URL: <https://learncpp.com>
4. C++ Documentation - офіційна документація та посібники від Microsoft. *Microsoft Learn*. URL: <https://learn.microsoft.com/cpp>
5. ISO C++ - Standard C++ Foundation. Офіційний сайт стандарту, статті та новини URL: <https://isocpp.org>

7. Регуляції і політики курсу

Відвідування занять. Регуляція пропусків.

Відвідування усіх занять є обов'язковим. Студенти зобов'язані дотримуватися усіх строків, визначених для виконання усіх видів робіт, передбачених даною дисципліною. Пропуски та запізнення на заняття є недопустимими.

Політика академічної доброчесності

Кожний студент зобов'язаний дотримуватися принципів академічної доброчесності. Письмові завдання з використанням часткових або повнотекстових запозичень з інших робіт без зазначення авторства – це *плагіат*. Використання будь-якої інформації (текст, фото, ілюстрації тощо) мають бути правильно процитовані з посиланням на автора! Якщо ви не впевнені, що таке плагіат, фабрикація, фальсифікація, порадьтеся з викладачем. До студентів, у роботах яких буде виявлено списування, плагіат чи інші прояви недоброчесної поведінки можуть бути застосовані різні дисциплінарні заходи (див. посилання на Кодекс академічної доброчесності ЗНУ в додатку до силабусу). Неприпустиме складання роботи, виконаної іншою особою.

Використання комп'ютерів/телефонів на занятті

Використання мобільних телефонів, ноутбуків та інших гаджетів під час лекційних та лабораторних занять дозволяється виключно у навчальних цілях (з активованим режимом «без звуку»).

Комунікація

Комунікація викладача зі студентами здійснюється безпосередньо на заняттях та додатково за допомогою месенджерів (наприклад, Telegram), електронної пошти і в СЕЗН Moodle (форум курсу, приватні повідомлення)

ДОДАТКОВА ІНФОРМАЦІЯ

ГРАФІК ОСВІТНЬОГО ПРОЦЕСУ НА 2025-2026 н.р. доступний за адресою:
<https://surl.li/vlwelj>

НАВЧАННЯ ТА ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ОСВІТИ. Перевірка набутих студентами знань, навичок та вмінь є невід'ємною складовою системи забезпечення якості освіти і проводиться відповідно до Положення про організацію та методику проведення поточного та підсумкового семестрового контролю навчання студентів Запорізького національного університету:
<https://surl.li/wdzjrl>

ПОВТОРНЕ ВИВЧЕННЯ ДИСЦИПЛІН. Наявність академічної заборгованості до 6 навчальних дисциплін (у тому числі проходження практики чи виконання курсової роботи) за результатами однієї екзаменаційної сесії є підставою для надання студенту права на повторне вивчення зазначених навчальних дисциплін. Процедура повторного вивчення визначається Положенням про порядок повторного вивчення навчальних дисциплін та повторного навчання у ЗНУ: <https://surl.lu/hfjbya>

ВИРІШЕННЯ КОНФЛІКТІВ. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, регламентуються Положенням про порядок і процедури вирішення конфліктних ситуацій у ЗНУ:
<https://surl.li/qgacqa>

Конфліктні ситуації, що виникають у сфері стипендіального забезпечення здобувачів вищої освіти, вирішуються стипендіальними комісіями факультетів, коледжів та університету в межах їх повноважень, відповідно до:

Положення про порядок призначення і виплати академічних стипендій у ЗНУ:
<https://surl.li/unwzzm>

Положення про призначення та виплату соціальних стипендій у ЗНУ:
<https://surl.lu/xkxmuz>

ПСИХОЛОГІЧНА ДОПОМОГА. Кабінет практичного психолога **Марті Ірини Вадимівни** – навч. корп. №4, каб. №235 (понеділок, середа, четвер 9.00-11.00, 13.00-15.00), навч. корп. №9 (ІННІ) каб.57 (п'ятниця 9.00-11.00, 13.00-15.00), гуртожиток №6 (вул. Добролюбова, 19, середа 9.00-11.00, 13.00-15.00). Попередній запис за тел.: 228-76-48, (099) 253-78-73 щоденно з 9 до 15.

УПОВНОВАЖЕНА ОСОБА З ПИТАНЬ ЗАПОБІГАННЯ ТА ВИЯВЛЕННЯ КОРУПЦІЇ Запорізького національного університету: **Банах Віктор Аркадійович**

Електронна адреса: v_banakh@znu.edu.ua



Гаряча лінія: тел. (061) 227-12-76, факс 227-12-88

РІВНІ МОЖЛИВОСТІ ТА ІНКЛЮЗИВНЕ ОСВІТНЄ СЕРЕДОВИЩЕ.

Центральні входи усіх навчальних корпусів ЗНУ обладнані пандусами для забезпечення доступу осіб з інвалідністю та інших маломобільних груп населення. Допомога для здійснення входу у разі потреби надається черговими охоронцями навчальних корпусів. Спеціалізована допомога: (061) 228-75-11 (начальник охорони). Порядок супроводу (надання допомоги) осіб з інвалідністю та інших маломобільних груп населення у ЗНУ: <https://surl.li/ivcwih>

РЕСУРСИ ДЛЯ НАВЧАННЯ

НАУКОВА БІБЛІОТЕКА: <http://library.znu.edu.ua>. Графік роботи абонементів: понеділок-п'ятниця з 08.00 до 16.00; вихідні дні: субота і неділя.

СИСТЕМА ЕЛЕКТРОННОГО ЗАБЕЗПЕЧЕННЯ НАВЧАННЯ ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ (СЕЗН ЗНУ):

<https://moodle.znu.edu.ua>.

Посилання для відновлення паролю:
<https://moodle.znu.edu.ua/mod/page/view.php?id=133015>.

ЦЕНТР ІНТЕНСИВНОГО ВИВЧЕННЯ ІНОЗЕМНИХ МОВ:

<http://sites.znu.edu.ua/child-advance/>