

СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «ОСНОВИ ПРОГРАМУВАННЯ»

Лектор курсу			Гавор Артур Станіславович, Старший викладач.		Контактна інформація лектора (e-mail), сторінка курсу в Google Classroom		e-mail: a.havor@duikt.edu.ua сторінка курсу в Google Classroom – https://classroom.google.com/c/Nzc0OTIxMjgyMTY0?cjc=hgehmwfq	
Галузь знань			F «Інформаційні технології»		Рівень вищої освіти		бакалавр	
Спеціальність			F2 Інженерія програмного забезпечення		Семестр		1	
Освітня програма			Технології цифрового розвитку		Тип дисципліни		Обов’язкова	
Обсяг:	Кредитів ECTS	Годин	За видами занять:					
			Лекцій	Семінарських занять	Практичних занять	Лабораторних занять	Самостійна підготовка	
	6	180	18	-	54	36	72	

АНОТАЦІЯ КУРСУ

Взаємозв'язок у структурно-логічній схемі

Освітні компоненти, які передують вивченню		Базова
Освітні компоненти для яких є базовою		Алгоритми і структури даних, Програмування на мові Python
Мета курсу:	Метою навчальної дисципліни «Основи програмування» є розвинення у студентів навичок складання програм мовою C++ для вирішення прикладних задач різного характеру.	

Компетентності відповідно до освітньої програми

Soft- skills / Загальні компетентності (ЗК)	Hard-skills / Спеціальні компетентності (ПП)
ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.	ПП1. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення. ПП3. Здатність розробляти архітектури, модулі та компоненти програмних систем. ПП7. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних ПП11. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення. ПП14. Здатність до алгоритмічного та логічного мислення.

Програмні результати навчання (ПРН)

ПРН6. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.
ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ОРГАНІЗАЦІЯ НАВЧАННЯ (1 семестр)

Тема, опис теми	Вид заняття	Оцінювання за тему	Форми і методи навчання/питання до самостійної роботи
Тема 1. Основи синтаксису та семантики мови C++. Змінні, типи, вирази і присвоювання. Основи ведення/виведення інформації.			

Рекомендовані джерела: 1, 2, 3, 7, 8, 9

Заняття 1.1. Основні поняття мови програмування C++. Парадигми програмування. Фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення, стандарти мови C++, сфери застосування мови C++. Змінні, типи, вирази. Властивості типів даних та базові типи в C++. Правила побудови виразів, використання операцій відповідно до типу даних; поняття сумісності типів. Поняття сумісності типів.	Лекція 1 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 1.2. Змінні, типи, вирази і присвоювання. Основні типи даних C++.	Практичне заняття 1 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач: складання простої програми мовою C++, використання змінних необхідних типів відповідно до вимог користувацької задачі; побудова математичних, логічних та інших виразів, потрібних для вирішення задач;
Заняття 1.3. Опис арифметичних, логічних і строкових виразів	Практичне заняття 2 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 1.4. Особливості введення/виведення даних в C++	Практичне заняття 3 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач застосування засобів введення/виведення C++ для організації діалогу з користувачем.
Заняття 1.5. Змінні, типи, вирази і присвоювання. Модифікатори базових типів в C++.	Лабораторне заняття 1 2 год.	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 1.6. Особливості введення/виведення даних в C++. Тип даних та змінних	Лабораторне заняття 2 год.	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач застосування засобів введення/виведення C++ для організації діалогу з користувачем.

Тема 2. Оператори розгалуження та вибору. Побітові операції. Оператори циклів.**Рекомендовані джерела: 1, 2, 3, 7, 8, 9**

Заняття 2.1. Види, синтаксис та порядок виконання операторів розгалуження та вибору в C++. Умовний оператор – синтаксис, порядок виконання, повний та неповні оператори розгалуження. Тернарний умовний оператор – синтаксис та особливості використання. Оператори циклів. Види циклів в C++. Синтаксис циклів, особливості роботи, приклади використання.	Лекція 2 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
---	-------------------	--	---

Заняття 2.2. Особливості застосування операторів розгалуження та оператору вибору.	Практичне заняття 4 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду типових задач на використання операторів розгалуження в C++.
Заняття 2.3. Вирішення задач із використанням операторів циклу. Обробка послідовностей.	Практичне заняття 5 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки послідовностей.
Заняття 2.4. Вирішення задач із використанням операторів циклу, розгалужень та побітових операцій. Обробка послідовностей.	Практичне заняття 6 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки послідовностей.
Заняття 2.5. Вирішення задач із використанням побітових операторів та розгалужень.	Лабораторне заняття 3 2 год	1 бал	Усне опитування, індивідуальне вирішення задач із використанням побітових операторів та розгалужень.
Заняття 2.6. Обробка послідовностей.	Лабораторне заняття 4 2 год	1 бал	Усне опитування, індивідуальне вирішення типових задач за темою заняття.

Тема 3. Масиви. Основні методи та типові задачі обробки масивів.

Рекомендовані джерела: 1, 2, 3, 7, 8, 9

Заняття 3.1 Класифікація складних структур даних у C++. Поняття масиву, види масивів, особливості представлення масивів у пам'яті C++. Одновимірні масиви: синтаксис опису, ініціалізація, доступ та обробка елементів. Лінійні та нелінійні схеми обробки одновимірних масивів (повний перебір, вставка, видалення, зсув елементів). Вибір стратегії та алгоритму розв'язання задач обробки масивів (зведення задачі до вже відомої, розділення на підзадачі). Типові задачі обробки одновимірних масивів. Багатовимірні масиви – синтаксис, представлення в пам'яті, особливості обробки. Обговорення типових задач і завдань для самостійного опрацювання.	Лекція 3 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 3.2. Основні методи та типові задачі обробки одновимірних масивів. Лінійна схема обробки.	Практичне заняття 7	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки одновимірних масивів.
Заняття 3.3. Основні методи та типові задачі обробки одновимірних масивів. Нелінійні схеми обробки.	Практичне заняття 8 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки одновимірних масивів.
Заняття 3.4. Основні методи та типові задачі обробки двовимірних масивів.	Практичне заняття 9 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки двовимірних масивів.

Заняття 3.5. Обробка одновимірних масивів.	Лабораторне заняття 5 2 год	2 бали	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 3.6. Обробка двовимірних масивів.	Лабораторне заняття 6 2 год	2 бали	Усне опитування, індивідуальне вирішення задач за темою заняття.
Тема 4. Посилання. Вказівники. Реалізація динамічних масивів. <u>Рекомендовані джерела:</u> 1, 2, 3, 7, 8, 9			
Заняття 4.1 Поняття вказівника та посилання в C++. Особливості оголошення та типові операції з вказівниками та посиланнями в C++. Використання вказівників в C++ для представлення різних структур даних. Обговорення задач для самостійного опрацювання. Динамічні масиви. Використання вказівників в C++ для представлення одновимірних, двовимірних та багатовимірних масивів. Проблеми витоку пам'яті. Особливості використання операторів роботи з пам'яттю при роботі з динамічними масивами. Зубчасті масиви. Обговорення задач для самостійного опрацювання.	Лекція 4 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 4.2. Посилання. Вказівники.	Практичне заняття 10 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач обробки посилання та вказівників, в тому числі виділення динамічної пам'яті.
Заняття 4.4. Вказівники. Реалізація одновимірних динамічних масивів.	Практичне заняття 11 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач на використання вказівників, в тому числі, при роботі з динамічними масивами.
Заняття 4.5. Вказівники. Реалізація двовимірних та багатовимірних динамічних масивів.	Практичне заняття 12 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення типових задач на використання вказівників, в тому числі, при роботі з динамічними масивами.
Заняття 4.6. Посилання. Вказівники.	Лабораторне заняття 7 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 4.7. Вказівники. Реалізація динамічних масивів.	Лабораторне заняття 8 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.

Тема 5. Рядки та стандартні бібліотеки STL. Функції C++ синтаксис та передача параметрів. Структури struct обробка даних і структуризація користувацьких бібліотек та модулів.

Рекомендовані джерела: 1, 2, 3, 7, 8, 9

Заняття 5.1. Представлення текстової інформації в C++, символіні та текстові літерали, C-style рядки, масиви символів, введення та виведення рядкових даних. Основні функції та методи обробки рядків. Використання контейнерів та структур STL для обробки текстових і одновимірних даних. Функції в C++, призначення, синтаксис, виклик функцій, повернення результату, передача параметрів за значенням. Структури struct у C++: опис і використання, звернення до полів, представлення гетерогенних даних, вкладені структури, складні структури даних на основі struct, базові елементи об'єктно-орієнтованого підходу. Структурування програм: користувацькі бібліотеки, багатофайлові проекти, директиви препроцесора (#pragma once, #define), принципи організації та оформлення коду. Обговорення задач для самостійного опрацювання.	Лекція 5 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 5.2. Використання контейнерів даних стандартної бібліотеки STL. Створення простих користувацьких функцій.	Практичне заняття 13 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення користувацьких задач з застосуванням функцій із контейнерами даних стандартної бібліотеки STL.
Заняття 5.3. Вирішення задач з використанням користувацьких функцій.	Практичне заняття 14 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення користувацьких задач з застосуванням функцій.
Заняття 5.4. Вирішення задач із обробки контейнерів даних та гетерогенних даних із використанням структур struct	Практичне заняття 15 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішуються типові задачі на обробку рядків за допомогою алгоритмів обробки одновимірних масивів, а також задачі з використанням структур struct для роботи з гетерогенними даними.
Заняття 5.5. Обробка контейнери даних стандартної бібліотеки STL з використанням користувацьких функцій.	Лабораторне заняття 9 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття
Заняття 5.6. Вирішення задач із обробки контейнерів даних та гетерогенних даних із використанням структур struct.	Лабораторне заняття 10 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття

Тема 6. Використання функцій в структурах struct. Елементи об'єктно-орієнтованого підходу. Інкапсуляції. Рекурсія. Рекурсивні функції.

Рекомендовані джерела: 1, 4, 5, 6, 7, 9

Заняття 6.1 Користувацькі бібліотеки та багатофайлові програми. Структурування коду з використанням функцій і бібліотек. Функції в структурах struct: внутрішні та зовнішні функції, підтримка елементів об'єктно-орієнтованої парадигми, конструктори та деструктори. Поняття рекурсії. Базис рекурсії, рекурентні вирази, реалізація рекурсивних функцій, дерево рекурсії, помилки нескінченної рекурсії та переповнення стеку. Поняття, інкапсуляції. Реалізація механізму інкапсуляції, особливості доступу до даних та методів struct. Обговорення задач для самостійного опрацювання.	Лекція 6 2 год		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 6.4. Обробка даних з використанням структур. Застосування механізму інкапсуляції.	Практичне заняття 16 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення користувацьких задач з використанням класів C++ із застосуванням механізму наслідування.
Заняття 6.5. Реалізація рекурсивних алгоритмів з використанням функцій в C++	Практичне заняття 17 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення окремих рекурсивних задач.
Заняття 6.6. Використання функцій в структурах struct.	Практичне заняття 18 2 год	2 бали	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення задач з обробки гетерогенних даних із використанням структур struct та функцій.
Заняття 6.7. Реалізація рекурсивних алгоритмів з використанням функцій в C++	Лабораторне заняття 11 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 6.8. Використання функцій в структурах struct.	Лабораторне заняття 12 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття шляхом структурування попередньо створених програм з обробки гетерогенних даних із використанням користувацьких функцій.
Тема 7. Ключові елементи об'єктно-орієнтованої парадигми програмування. Базові поняття. <u>Рекомендовані джерела:</u> 1, 4, 5, 6, 7, 9			
Заняття 7.1 Ключові елементи об'єктно-орієнтованої парадигми програмування у класах. Базові поняття поняття класу, об'єкту; атрибути та методи класу. Поняття конструктора та деструктора. Види конструкторів. Обговорення задач для самостійного опрацювання.	Лекція 7		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 7.2. Обробка даних з використанням класів C++	Практичне заняття 19 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення користувацьких задач з використанням класів C++

Заняття 7.3. Обробка даних з використанням класів C++	Практичне заняття 20 2 год	2 бали	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 7.4. Обробка даних з використанням класів C++	Практичне заняття 21 2 год	2 бали	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 7.5. Обробка даних з використанням класів C++	Лабораторне заняття 13 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 7.6. Обробка даних з використанням класів C++	Лабораторне заняття 14 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Тема 8. Об'єктно Орієнтована парадигма програмування. <u>Рекомендовані джерела:</u> 1, 4, 5, 6, 7, 9			
Заняття 8.1 Поняття наслідування, інкапсуляції, поліморфізму Квантори видимості. Реалізація механізму наслідування, особливості доступу до даних та методів в класах-нащадках. Обговорення задач для самостійного опрацювання.	Лекція 8		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 8..2. Обробка даних з використанням класів C++. Застосування механізму наслідування.	Практичне заняття 22 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, групові заняття із розгляду та вирішення користувацьких задач з використанням класів C++ із застосуванням механізму наслідування.
Заняття 8.3. Застосування механізму абстракції при обробці даних на основі класів C++	Практичне заняття 23 2 год	2 бали	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 8.4. Обробка даних з використанням класів C++. Застосування механізму наслідування.	Практичне заняття 24 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 8.5. Застосування механізму абстракції при обробці даних на основі класів C++	Лабораторне заняття 15 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.
Заняття 8.6. Застосування механізму абстракції при обробці даних на основі класів C++	Лабораторне заняття 16 2 год	1 бал	Усне опитування, індивідуальне вирішення задач за темою заняття.

Тема 9. Абстракції та інтерфейси. Базові патерни проектування**Рекомендовані джерела: 1, 4, 5, 6, 7, 9**

Заняття 9.1. Поняття абстракції та інтерфейсів. Абстрактні класи та чисто віртуальні функції. Поліморфізм та наслідування. Квантори доступу. Огляд базових патернів проектування.	Лекція 9		Пояснювально-ілюстративний, лекція-візуалізація, навчальна дискусія
Заняття 9.2. Абстракція та інтерфейси в C++. Абстрактні класи	Практичне заняття 25 2 год	1 бал	Усне опитування по матеріалу лекції, навчальна дискусія, розв'язання задач з використанням абстрактних класів та чисто віртуальних функцій
Заняття 9.3. Поліморфізм та наслідування при роботі з інтерфейсами	Практичне заняття 26 2 год	2 бали	Усне опитування, групове та індивідуальне вирішення задач із застосуванням наслідування та поліморфізму
Заняття 9.4. Застосування базових патернів проектування	Практичне заняття 27 2 год	1 бал	Усне опитування, аналіз прикладів та вирішення задач із використанням базових патернів проектування
Заняття 9.5. Реалізація абстракцій та інтерфейсів у програмах C++	Лабораторне заняття 17 2 год	1 бал	Усне опитування, індивідуальна розробка програм з використанням абстрактних класів та інтерфейсів
Заняття 9.6. Застосування патернів проектування в об'єктно-орієнтованих програмах	Лабораторне заняття 18 2 год	1 бал	Усне опитування, індивідуальне виконання лабораторних завдань із застосуванням базових патернів проектування

Самостійна робота

Використання базових операторів для обробки послідовностей.	7 год		Виконання завдань для самостійної роботи
Задачі обробки масивів різної складності.	10 год		Виконання завдань для самостійної роботи
Робота зі структурами struct.	7 год		Виконання завдань для самостійної роботи
Структурування програм із використанням функцій та бібліотек.	14 год.		Виконання завдань для самостійної роботи
Рекурсивні алгоритми	8 год.		Виконання завдань для самостійної роботи
Робота з пам'яттю	8 год.		Виконання завдань для самостійної роботи
Перевантаження операторів	8 год.		Виконання завдань для самостійної роботи
Використання ключових елементів об'єктно-орієнтованої парадигми програмування.	10 год.		Виконання завдань для самостійної роботи

МАТЕРІАЛЬНО-ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

● Мультимедійний проектор.
● Комп'ютерний клас для проведення практичних та лабораторних занять.
● Програмне забезпечення: Visual Studio, онлайн компілятори C++.

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

1. International Standard ISO/IEC 14882:2014(E) – Programming Language C++ , ISBN-13: 978- 0321563842: [Електронний ресурс]. – Режим доступу: <https://isocpp.org/std/the-standard>.
2. Шушупа О.М., Золотухіна О.А. Логіка та алгоритми обробки даних. Навчальний посібник. – Київ: ДУТ, 2019. – 121 с.
3. C++ Crash Course: A Fast-Paced Introduction./ Lospinoso Josh. ISBN 1593278885. - 2019.- 792с.
4. C/C++ language and standard libraries reference: [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/en-us/library/hh875057.aspx>
5. Jeremy A. Hansen. The Rook's Guide to C++. Rook's Guide Press. URL <https://rooksguide.files.wordpress.com/2013/12/rooks-guide-isbn-version.pdf>
6. Золотухіна О.А., Програмування C++. Навчальний посібник. Частина 2. К.: ДУТ, 2020. 52 с.
7. Програмування на C++ в прикладах і задачах : Навч. посіб. / О. Васильєв. – Київ : Видавництво Ліра-К, 2017. – 382 с.
8. www.netacad.com, курс CPA - Programming Essentials in C++.
9. <https://classroom.google.com/c/Nzc0OTIxMjgyMTY0?cjc=hgehmwfg> – сторінка курсу в Google Classroom.

ПОЛІТИКА КУРСУ («ПРАВИЛА ГРИ»)

- Курс передбачає самостійне роботу та роботу в колективі.
- Освоєння дисципліни передбачає обов'язкове відвідування лекцій, практичних та лабораторних занять, а також самостійну роботу.
- Самостійна робота включає в себе теоретичне вивчення питань, що стосуються тем лекційних занять, які не ввійшли в теоретичний курс, або ж були розглянуті коротко, їх поглиблена проробка за рекомендованою літературою.
- Спілкуючись з учасниками навчального процесу, студенти мають дотримуватися етичних норм, утримуватися від гучних проявів емоцій, бути політично та гендерно коректними й толерантними, поважати звичаї й традиції різних етнічних, культурних, соціальних груп і релігійних конфесій. Під час лекційних і практичних, лабораторних занять студентам забороняється приймати їжу та користуватися мобільним телефоном. Використання мобільних телефонів допускається при виконанні практичних та лабораторних занять для доступу до навчальних та довідкових матеріалів.
- Усі завдання, передбачені програмою, мають бути виконані у встановлений термін, що залежить від розкладу навчальної групи.
- Якщо студент із поважних причин був відсутній на практичному чи лабораторному занятті, він має право його відпрацювати. Відпрацювання полягає у виконанні студентом в позанавчальний час завдання за темою заняття та захисті роботи викладачеві на наступному лабораторному/практичному занятті (при наявності вільного часу у викладача), або під час додаткової консультації з викладачем, час якої узгоджується окремо. Для виконання завдань курсу може використовуватись обладнання лабораторій кафедри Технологій цифрового розвитку або власна комп'ютерна техніка студента.
- Під час вивчення дисципліни всі учасники навчального процесу мають дотримуватися вимог, установлених Положенням про академічну доброчесність здобувачів освіти у ДУІКТ, своєю діяльністю утверджувати академічну доброчесність як засадничу цінність університетської спільноти. Студенти не повинні отримувати чи надавати недозволеної допомоги, вдаватися до жодних інших нечесних дій під час виконання навчальних завдань. У разі виявлення факту плагіату студент отримує за завдання 0 балів.
- За використання телефонів і комп'ютерних засобів без дозволу викладача, порушення дисципліни студент видаляється з заняття, за заняття отримує 0 балів.
- Кафедра інженерії програмного забезпечення ДУІКТ всіляко сприятиме вирішенню індивідуальних проблем студентів з особливими потребами, сприймаючи їх як рівних в інтегрованому студентському колективі.

КРИТЕРІЙ ТА МЕТОДИ ОЦІНЮВАННЯ

Оцінювання студентів в кожному семестрі здійснюється за накопичувальною 100-бальною системою, складається із двох основних блоків і розподіляється в пропорції 60 (бали напрацьовані під час вивчення дисципліни) на 40 (екзамен). Максимальна кількість балів за видами поточного контролю за семестр становить 120 балів – ця сума нормується до 60 балів. Розрахунок рейтингових балів за видами поточного та підсумкового контролю за кожний семестр здійснюється за формулою $\text{Рейтинговий бал} = K \cdot \Pi + I + D$, де

Коефіцієнт $K = 60/120$;

Поточний контроль студента (Π) - сума всіх балів за семестр;

Іспит (мак) (I) – 40;

Додаткові бали (D) - визначаються в залежності від виду виконаної роботи.

Якщо сумарна кількість балів з урахуванням додаткових балів перевищує 100, вона округлюється до 100.

Семестровий контроль в кожному з семестрів – іспит (тести з автоматизованою перевіркою та практичні задачі, які перевіряються викладачем), максимальна оцінка – 40 балів.

Тестова частина екзаменаційного білету кожного семестру містить завдання різної складності, бал за кожне завдання варіюється від 0.5 до 2 в залежності від складності завдання. Сумарна кількість балів за тестову частину складає 15 балів. Практична частина екзамену передбачає написання студентом програмного коду для вирішення задач за темами семестру. Практична частина сумарно оцінюється в 25 балів.

Під час екзамену при виконанні практичної частини завдання дозволяється користуватися конспектом лекцій та друкованими навчально-методичними матеріалами. Під час тестування заборонено використання будь яких зовнішніх джерел. Використання ресурсів мережі Інтернет, Chat GPT та інших систем на основі штучного інтелекту під час практичної частини екзамену заборонено.

Умовою допуску до екзамену є набір критичної мінімальної кількості балів з усіх практичних, лабораторних робіт і виконання самостійних завдань, які передбачені структурною освітньої компоненти Програмування C++, що становить 40 балів.

Якщо студента не допущено до складання екзамену, як такого, що не виконав індивідуальний план, йому надається час до перескладання для виконання всіх вимог допуску.

Якщо студент був допущений до складання екзамену, але не з'явився без поважної причини, то вважається, що він використав першу спробу скласти екзамен і має заборгованість.

Студент має право на два перескладання. При повторному перескладанні екзамену його у студента може приймати комісія, яка створюється директором ННІТ. Оцінка комісії є остаточною.

Бали, отримані впродовж семестру, додаються до отриманих під час екзамену.

Форми контролю	Види навчальної роботи	Оцінювання
ПОТОЧНИЙ КОНТРОЛЬ	- Виконання лабораторних робіт, - Виконання практичних робіт	60 балів (нормована оцінка)
ПІДСУМКОВЕ ОЦІНЮВАННЯ <i>Іспит</i>	- Письмова екзаменаційна робота алгоритмічне програмування - Письмова екзаменаційна робота функціональне програмування - Письмова екзаменаційна робота ООП програмування	40 балів (10 балів + 10 балів + 20 балів)

Додаткова оцінка

Види навчальної роботи	Оцінювання
Самостійне вирішення задач, які не входять до обов'язкового розглядання та наведені в лекціях частково.	Не більше 10 задач (за умови виконання основного плану). Визначається диференційовано відповідно до складності виконаного завдання. Максимально до 3 балів за 1 завдання.
Участь у олімпіадах з програмування.	Кількість олімпіад не обмежується. До 10 балів в залежності від результатів участі.
Отримання сертифікату з курсу CPA - Programming Essentials in C++ або подібних курсів	Не більше 2 курсів за семестр. До 10 балів за курс. Якщо теми курсів перетинаються між собою, то зарахування балів відбувається лише один раз. Враховуються бали лише за ті теми курсів, що вказані в силабусі дисципліни.
Виконання додаткових творчих завдань (наприклад, робота над індивідуальним чи груповим проектом, участь в стартапі тощо)	Визначається диференційовано відповідно до складності виконаного завдання. Додаткові творчі завдання у форматі індивідуального чи групового проекту, стартапу

		тощо можуть змінювати основні роботи (практичні, самостійні), якщо вони покривають практичну складову відповідної теми освітнього компоненту.	
Студент може отримати додаткові бали протягом семестру. Кількість балів за додаткові види робіт визначається диференційовано відповідно до виду та складності виконаного завдання. Максимальна кількість додаткових балів, які можуть бути зараховані здобувачу освіти - 10 балів.			
ПІДСУМКОВА ОЦІНКА ЗА ДИСЦИПЛІНУ			
бали	Критерії оцінювання	Рівень компетентності	Оцінка /запис в екзаменаційній відомості
90-100	Студент демонструє повні й міцні знання навчального матеріалу в обсязі, що відповідає робочій програмі дисципліни, правильно й обґрунтовано приймає необхідні рішення в різних нестандартних ситуаціях. Вміє реалізувати теоретичні положення дисципліни в практичних розрахунках, аналізувати та співставляти дані об'єктів діяльності фахівця на основі набутих з даної та суміжних дисциплін знань та умінь. Знає сучасні технології та методи розрахунків з даної дисципліни. За час навчання при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань проявив вміння самостійно вирішувати поставлені завдання, активно включатись в дискусії, може відстоювати власну позицію в питаннях та рішеннях, що розглядаються. Зменшення 100-бальної оцінки може бути пов'язане з недостатнім розкриттям питань, що стосується дисципліни, яка вивчається, але виходить за рамки об'єму матеріалу, передбаченого робочою програмою, або студент проявляє невпевненість в тлумаченні теоретичних положень чи складних практичних завдань.	Високий Повністю забезпечує вимоги до знань, умінь і навичок, що викладені в робочій програмі дисципліни. Власні пропозиції студента в оцінках і вирішенні практичних задач підвищує його вміння використовувати знання, які він отримав при вивченні інших дисциплін, а також знання, набуті при самостійному поглибленому вивченні питань, що відносяться до дисципліни, яка вивчається.	Відмінно / Зараховано (А)
82-89	Студент демонструє гарні знання, добре володіє матеріалом, що відповідає робочій програмі дисципліни, робить на їх основі аналіз можливих ситуацій та вміє застосовувати теоретичні положення при вирішенні практичних задач, але допускає окремі неточності. Вміє самостійно виправляти допущені помилки, кількість яких є незначною. Знає сучасні технології та методи розрахунків з даної дисципліни. За час навчання при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань та поясненні прийнятих рішень, дає вичерпні пояснення.	Достатній Забезпечує студенту самостійне вирішення основних практичних задач в умовах, коли вихідні дані в них змінюються порівняно з прикладами, що розглянуті при вивченні дисципліни	Добре / Зараховано (В)
75-81	Студент в загальному добре володіє матеріалом, знає основні положення матеріалу, що відповідає робочій програмі дисципліни, робить на їх основі аналіз можливих ситуацій та вміє застосовувати при вирішенні типових практичних завдань, але допускає окремі неточності. Вміє пояснити основні положення виконаних завдань та дати правильні відповіді при зміні результату при заданій зміні вихідних параметрів. Помилки у відповідях/ рішеннях/ розрахунках не є системними. Знає характеристики основних положень, що мають визначальне значення при проведенні практичних занять, при виконанні індивідуальних / контрольних завдань та поясненні прийнятих рішень, в межах дисципліни, що вивчається.	Достатній Конкретний рівень, за вивченим матеріалом робочої програми дисципліни. Додаткові питання про можливість використання теоретичних положень для практичного використання викликають утруднення.	Добре / Зараховано (С)

67-74	Студент засвоїв основний теоретичний матеріал, передбачений робочою програмою дисципліни, та розуміє постанову стандартних практичних завдань, має пропозиції щодо напрямку їх вирішень. Розуміє основні положення, що є визначальними в курсі, може вирішувати подібні завдання тим, що розглядалися з викладачем, але допускає значну кількість неточностей і грубих помилок, які може усувати за допомогою викладача. .	Середній Забезпечує достатньо надійний рівень відтворення основних положень дисципліни	Задовільно / Зараховано (D)
60-66	Студент має певні знання, передбачені в робочій програмі дисципліни, володіє основними положеннями, що вивчаються на рівні, який визначається як мінімально допустимий. З використанням основних теоретичних положень, студент з труднощами пояснює правила вирішення практичних/розрахункових завдань дисципліни. Виконання практичних / індивідуальних / контрольних завдань значно формалізовано: є відповідність алгоритму, але відсутнє глибоке розуміння роботи та взаємозв'язків з іншими дисциплінами.	Середній Є мінімально допустимим у всіх складових навчальної програми з дисципліни	Задовільно / Зараховано (E)
40-59	Студент може відтворити окремі фрагменти з курсу. Незважаючи на те, що програму навчальної дисципліни студент виконав, працював він пасивно, його відповіді під час практичних робіт в більшості є невірними, необґрунтованими. Цілісність розуміння матеріалу з дисципліни у студента відсутні.	Низький Не забезпечує практичної реалізації задач, що формуються при вивченні дисципліни	Незадовільно з можливістю повторного складання) / Не зараховано (FX) В залікову книжку не представляється
0-39	Студент повністю не виконав вимог робочої програми навчальної дисципліни. Його знання на підсумкових етапах навчання є фрагментарними. Студент не допущений до здачі екзамену/заліку.	Незадовільний Студент не підготовлений до самостійного вирішення задач, які окреслює мета та завдання дисципліни	Незадовільно з обов'язковим повторним вивченням / Не допущений (F) В залікову книжку не представляється

ПОЛІТИКА ДОБРОЧЕСНОСТІ

Здобувач вищої освіти виконуючи самостійну або індивідуальну роботу повинен дотримуватись політики доброчесності. У разі наявності плагіату в будь-яких видах робіт Здобувача, він отримує незадовільну оцінку і повинен повторно виконати завдання, які передбачені у Силабусі.