



МІЖНАРОДНИЙ ЄВРОПЕЙСЬКИЙ УНІВЕРСИТЕТ

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
«ЄВРОПЕЙСЬКА ШКОЛА ІТ»**

ЗАТВЕРДЖЕНО
Директор ННІ
«Європейська школа бізнесу»
Юлія РЕМИГА
Від «11» 09 2023 р.
42947833

М.П.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ОСНОВИ ПРОГРАМУВАННЯ

(назва навчальної дисципліни/освітньої компоненти)


Рівень вищої освіти: перший (бакалаврський)
Ступінь вищої освіти: бакалавр
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 121 «Інженерія програмного забезпечення»

Київ – 2023

Робоча програма навчальної дисципліни «**Основи програмування**» складена на основі освітньо-професійної програми «Інженерія програмного забезпечення» для першого (бакалаврського) рівня спеціальності 121 «Інженерія програмного забезпечення», а також навчальних планів №121бд/23-09, №121бз/23-10 затверджених Вченою радою Університету «30» травня 2023 року, протокол № 4.

РОЗРОБНИК: Шерман Зоя Олександрівна кандидат фізико-математичних наук, доцент.

РЕЦЕНЗЕНТ:, Глазунов Микола Михайлович, доктор фізико-математичних наук, професор.

Гарант освітньої програми  Олександр НЕСТЕРЕНКО, доктор технічних наук, професор

Робочу програму навчальної дисципліни розглянуто та схвалено кафедрою інформаційних технологій, протокол № 1 від «31» серпня 2023 р.

Розглянуто і схвалено Вченою радою Навчально-наукового інституту «Європейська школа бізнесу», протокол № 1 від «11» вересня 2023 р.

ВСТУП

Програма вивчення навчальної дисципліни «Основи програмування» складена відповідно до Стандарту вищої освіти України (далі – Стандарт) галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення».

Опис навчальної дисципліни (анотація). Дана навчальна дисципліна є однією з дисциплін фахової підготовки майбутніх розробників програмного забезпечення.

Таблиця 1

Найменування показників	Галузь знань, напрям підготовки, освітній рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 4	Галузь знань, 12 «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ»	Нормативна	
Розділів – 1	Спеціальність: 121 «ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»	Рік підготовки	
Змістових розділів – 1		1	1
Індивідуальне науково-дослідне завдання:		Семестр	
		1-2	1-2
Загальна кількість годин – 120		Лекції	
		32	12
	Практично-лабораторні		
Тижневе навантаження: аудиторних – 3 самостійної роботи студента – 4	Освітній рівень: бакалавр	Самостійна робота	
		144	220
		Вид контролю:	
		Залік, екзамен	Залік, екзамен

Предметом навчальної дисципліни є теорія і практика застосування у програмуванні базових алгоритмічних структур і базових структур даних на базі сучасних технологій розробки програмного забезпечення.

Міждисциплінарні зв'язки: Навчальна дисципліна реалізує міждисциплінарні зв'язки з такими навчальними предметами, як: «Операційні системи та системи програмного забезпечення», «Обчислювальні системи та мережі».

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. **Метою** викладання навчальної дисципліни «Основи програмування» є формування у майбутніх фахівців знань з основних понять алгоритмізації і техніки застосування у програмуванні базових алгоритмічних структур (організація програм) і базових структур даних (організація даних).

1.2. Основними завданнями вивчення дисципліни «Основи програмування» є:

- основні етапи процесу проектування програмного забезпечення;
- типові алгоритмічні конструкції;
- принципи процедурного і структурованого програмування; особливості застосування сучасних базових інструментальних програмних засобів, призначених для вирішення задач пов'язаних з підготовкою екіпажів до польоту;
- базові типи даних;
- похідні типи даних: переліки, покажчики, посилання, масиви, структури, об'єднання;
- оператори управління програмою; команди передпроцесорної обробки;
- правила роботи з функціями;
- систему уведення-виведення C++;
- основні принципи роботи з файлами;
- основу побудови програм на керованому C++.

1.3. Компетентності та результати навчання, формуванню яких сприяє дисципліна (взаємозв'язок з нормативним змістом підготовки здобувачів вищої освіти, сформульованим у термінах результатів навчання у Стандарті).

Згідно з вимогами стандарту дисципліна забезпечує набуття студентами **компетентностей**:

Таблиця 2

<i>Інтегральна компетентність</i>	Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми, які характеризуються комплексністю і невизначеністю умов у сфері розробки програмного забезпечення.
<i>Загальні компетентності</i>	ЗК 01. Здатність до абстрактного мислення, аналізу та синтезу.
<i>Спеціальні (фахові, предметні) компетентності</i>	СК3. Здатність розробляти архітектури, модулі та компоненти програмних систем. СК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення. СК14. Здатність до алгоритмічного та логічного мислення

Деталізація компетентностей відповідно до дескрипторів НРК у формі «Матриці компетентностей» наведено у табл. 3.

Таблиця 3

Матриця компетентностей

№	Компетентність	Знання	Уміння / навички	Комунікація	Автономія та відповідальність
Інтегральна компетентність					
1.	Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми, які характеризуються комплексністю і невизначеністю умов, у сфері розробки програмного забезпечення.	Теорії побудови операційних систем та взаємодії з прикладним програмним забезпеченням	Використовувати інформаційні технології, основні системні та прикладні програми для вирішення практичних завдань	Програмна взаємодія	Самостійне проектування і апробація на виробництві
Загальні компетентності					
2.	ЗК 01. Здатність до абстрактного мислення, аналізу та синтезу	структуру операційних систем, загальних принципів їх функціонування	застосовувати операції взаємодії з середовищем ОС при розробці прикладних програм	Зв'язок теоретичних і практичних знань	Спостереження процесів обробки інформації
Спеціальні (фахові, предметні) компетентності					
3.	СК3. Здатність розробляти архітектуру, модулі та компоненти програмних систем. СК13. Здатність обґрунтовано обирати та освоювати інструментарій розробки супроводження програмного забезпечення. СК14. Здатність до алгоритмічного та логічного мислення	основ мультипрограмування віртуалізації та розподілених обчислень, відповідних правил та функцій прикладного програмування	використання програмних засобів в операційному середовищі	Застосовування технологій паралельної роботи	Опис інформаційних процесів

Інтегративні кінцеві програмні результати навчання, формуванню яких сприяє навчальна дисципліна:

Програмні результати навчання	ПР03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення. ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення
--------------------------------------	--

Результати навчання:

Після опанування дисципліни студент повинен

знати:

- основні етапи процесу проектування програмного забезпечення;
- типові алгоритмічні конструкції;
- принципи процедурного і структурованого програмування; особливості застосування сучасних базових інструментальних програмних засобів;
- базові типи даних;
- похідні типи даних: переліки, покажчики, посилання, масиви, структури, об'єднання;
- оператори управління програмою; команди передпроцесорної обробки;
- правила роботи з функціями;
- систему введення-виведення C++;
- основні принципи роботи з файлами;
- основу побудови програм на керованому C++;

уміти:

- використовувати засоби програмування, основні системні виклики для вирішення практичних завдань прикладного програмування;
- використовувати функції та бібліотеки програмних засобів для розробки прикладних програм;
- застосовувати технології роботи з сучасними базовими інструментальними програмними засобами.

2. ІНФОРМАЦІЙНИЙ ОБСЯГ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

На вивчення навчальної дисципліни «Основи програмування» відводиться 120 годин 4 кредити ЄКТС.

Тема 1. Вступ. Основні поняття та означення

Тема 2. Синтаксис та семантика мови C++

Тема 3. Структура програми на мові C++. Оголошення змінних

Тема 4. Керування потоками вводу-виводу. Вирази

Тема 5. Засоби мови C++ для реалізації основних структур алгоритмів.

Оператор if.

Тема 6. Оператори циклу. Цикл while.

Тема 7. Цикли do-while.

Тема 8 Цикли for.

- Тема 9. Складні типи даних в С++. Одновимірні масиви.
Тема 10. Обробка одновимірних масивів.
Тема 11. Складні типи даних в С++. Двовимірні масиви.
Тема 12. Обробка двовимірних масивів.
Тема 13. Функції. Вказівники та посилання
Тема 14. Обробка рядків
Тема 15. Робота з контейнерами std: vector, list.
Тема 16. Робота з контейнерами std: deque, stack, queue.
Тема 17. Робота з контейнерами std:map, multimap.
Тема 18. Файловий ввід/вивід. Робота з eof.

3. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістових модулів і тем	Кількість годин				
	усього	у тому числі			
		л	п/з	лаб	СРС
1	2	3	4	5	6
Змістовий модуль 1. Основні поняття програмування.					
Тема 1. Вступ. Основні поняття та означення	4	1	2	-	1
Тема 2. Синтаксис та семантика мови С++	4	1	2	-	1
Тема 3. Структура програми на мові С++. Оголошення змінних	4	1	2	-	1
Тема 4. Керування потоками вводу-виводу. Вирази	4	1	2	-	1
Тема 5. Засоби мови С++ для реалізації основних структур алгоритмів. Оператор if.	5	2	2	-	1
Тема 6. Оператори циклу. Цикл while.	5	2	2	-	1
Разом за змістовим модулем	26	8	12	-	6
Змістовний модуль 2. Засоби реалізації основних структур в програмуванні.					
Тема 7. Цикли do-while.	12	2	4	-	1
Тема 8 Цикли for.	12	2	4	-	1

Тема 9. Складні типи даних в С++. Одновимірні масиви.	12	2	4	-	1
Тема 10. Обробка одновимірних масивів.	12	2	4	-	1
Тема 11. Складні типи даних в С++. Двовимірні масиви.	12	2	4	-	1
Тема 12. Обробка двовимірних масивів.	12	2	4	-	1
Разом за змістовим модулем	42	12	24	-	6
Змістовний модуль 3. Робота з функціями вводу/виводу та робота з строковими змінними.					
Тема 13. Функції. Вказівники та посилання	8	2	4	-	2
Тема 14. Обробка рядків	10	2	6	-	2
Тема 15. Робота з контейнерами std: vector, list.	8	2	4	-	2
Тема 16. Робота з контейнерами std: deque, stack, queue.	10	2	6	-	2
Тема 17. Робота з контейнерами std:map, multimap.	8	2	4	-	2
Тема 18. Файловий ввід/вивід. Робота з eof.	8	2	4	-	2
Разом за змістовим модулем	52	12	28	-	12
Усього годин	120	32	64		24

4. ТЕМИ ЛЕКЦІЙ

№ з/п	Назва теми (короткий зміст)	Кількість годин
1	Тема 1. Вступ. Основні поняття та означення Класифікація і особливостями сучасних мов програмування; середовища програмування та елементами вікна середовища програмування Microsoft Visual Studio;	1
2	Тема 2. Синтаксис та семантика мови С++ Множина символів С++ та множина представимих символів. Правила формування констант, ідентифікаторів. Ключові слова. Використання коментарів у програмах. Поняття лексеми.	1
3	Тема 3. Структура програми на мові С++. Оголошення змінних.	1

	<p>Склад програми на мові C++. Вихідні файли програми. Виконання програм. Особливості функції main(). Поняття "час життя" і "область видимості".</p> <p>Базові типи даних. Правила приведення базових типів даних. Модифікатори змінних.</p> <p>Автоматичні змінні. Реєстрові змінні. Зовнішні змінні та функції статичні змінні.</p>	
4	<p>Тема 4. Керування потоками вводу-виводу. Вирази. Основні оператори вводу та виводу на мові C++, зокрема на прикладі printf() та scanf(). Формат вводу-виводу різних типів даних (чисел, рядків, вказівників тощо).</p> <p>Арифметичні операції. Оператор присвоєння . Поняття виразу. Оператори інкремента і декремента. Оператор sizeof. Порозрядні логічні операції. Операції зсуву вліво і вправо. Оператори порівняння. Операція "кома". Пріоритет і порядок виконання операцій.</p>	1
5	<p>Тема 5. Засоби мови C++ для реалізації основних структур алгоритмів. Оператор if.</p> <p>Умовні оператори. Оператори if. Оператори if-else. Умовний оператор ?:. Оператор switch.</p>	2
6	<p>Тема 6. Оператори циклу. Цикл while.</p> <p>Поняття циклу: цикли while, do while, for. Управляючі оператори в циклах: оператор break, оператор continue.</p>	2
7	<p>Тема 7. Цикли do-while.</p> <p>Вкладені цикли. Інтегрований налагоджувач Microsoft Visual Studio.</p>	2
8	<p>Тема 8. Цикл for.</p> <p>Засоби опрацювання елементів циклу for.</p>	2
9	<p>Тема 9. Складні типи даних в C++. Одновимірні масиви. Поняття масиву даних. Оголошення, ініціалізація і виведення масивів. Типові алгоритми опрацювання масивів. Визначення максимального і мінімального елементів та їхніх номерів.</p>	2
10	<p>Тема 10. Обробка одновимірних масивів.</p> <p>Алгоритм упорядкування елементів одновимірного масиву. Інші методи впорядкування масивів.</p>	2
11	<p>Тема 11. Складні типи даних в C++. Двовимірні масиви. Ініціалізація двовимірних масивів. Типові задачі, в яких використовуються двовимірні масиви. Масиви структур. Показчики на структури. Передача по посиланню членів масивів структур. Об'єднання й операції з ними.</p>	2
12	<p>Тема 12. Обробка двовимірних масивів.</p> <p>Алгоритми обробки елементів двовимірного масиву.</p>	2
13	<p>Тема 13. Функції. Вказівники та посилання.</p> <p>Поняття про структурне програмування. Оголошення та виклик функцій. Передача аргументів. Прототипи функцій. Область видимості. Локальні і глобальні змінні. Аргументи за замовчуванням. Перенавантаження та шаблони функцій. Рекурсія.</p> <p>Загальний огляд. Разіменування вказівників. Арифметика вказівників. Вказівники. на вказівники. Вказівники. на</p>	2

	функції. Посилання. Передача параметрів за посиланням і за значенням. Використання вказівників і посилань із ключовим словом <code>const</code> .	
14	Тема 14. Обробка рядків Огляд класу <code>string</code> . Виконання базових операцій над об'єктами <code>string</code> . Робота з об'єктами <code>string</code> за допомогою ітераторів.	2
15	Тема 15. Робота з контейнерами <code>std: vector, list</code> . Виконання базових операцій над об'єктами <code>vector, list</code> . Робота з об'єктами <code>vector, list</code> за допомогою ітераторів.	2
16	Тема 16. Робота з контейнерами <code>std: deque, stack, queue</code> . Виконання базових операцій над об'єктами <code>deque, stack, queue</code> . Робота з об'єктами <code>deque, stack, queue</code> за допомогою ітераторів.	2
17	Тема 17. Робота з контейнерами <code>std:map, multimap</code> . Виконання базових операцій над об'єктами <code>map, multimap</code> . Робота з об'єктами <code>map, multimap</code> за допомогою ітераторів.	2
18	Тема 18. Файловий ввід/вивід. Робота з <code>eof</code> . Огляд вводу/виводу. Параметри й аргументи функцій. Аргументи за замовчуванням. Простір імен. Передача аргументів. Прототипи функцій. Область видимості. Перенавантаження та шаблони функцій. Рекурсія. Необхідні кроки роботи з <code>eof</code> . Варіанти та альтернативи.	2

5. ТЕМИ ЛАБОРАТОРНО-ПРАКТИЧНИХ ЗАНЯТЬ

№ з/п	Назва теми (короткий зміст)	Кількість годин
1	Тема 1. Вступ. Основні поняття та означення Класифікація і особливостями сучасних мов програмування; середовища програмування та елементами вікна середовища програмування Microsoft Visual Studio;	2
2	Тема 2. Синтаксис та семантика мови C++ Множина символів C++ та множина представимих символів. Правила формування констант, ідентифікаторів. Ключові слова. Використання коментарів у програмах. Поняття лексеми.	2
3	Тема 3. Структура програми на мові C++. Оголошення змінних. Склад програми на мові C++. Вихідні файли програми. Виконання програм. Особливості функції <code>main()</code> . Поняття "час життя" і "область видимості". Базові типи даних. Правила приведення базових типів даних. Модифікатори змінних. Автоматичні змінні. Реєстрові змінні. Зовнішні змінні та функції статичні змінні.	2
4	Тема 4. Керування потоками вводу-виводу. Вирази. Основні оператори вводу та виводу на мові C++, зокрема на прикладі <code>printf()</code> та <code>scan()</code> . Формат вводу-виводу різних типів даних (чисел, рядків, вказівників тощо).	2

	Арифметичні операції. Оператор присвоювання . Поняття виразу. Оператори інкремента і декремента. Оператор sizeof. Порозрядні логічні операції. Операції зсуву вліво і вправо. Оператори порівняння. Операція "кома". Пріоритет і порядок виконання операцій.	
5	Тема 5. Засоби мови C++ для реалізації основних структур алгоритмів. Оператор if. Умовні оператори. Оператори if. Оператори if-else. Умовний оператор ?: . Оператор switch.	2
6	Тема 6. Оператори циклу. Цикл while. Поняття циклу: цикли while, do while, for. Управляючі оператори в циклах: оператор break, оператор continue.	4
7	Тема 7. Цикли do-while. Вкладені цикли. Інтегрованої налагоджувач Microsoft Visual Studio.	4
8	Тема 8. Цикл for. Засоби опрацювання елементів циклу for.	4
9	Тема 9. Складні типи даних в C++. Одновимірні масиви. Поняття масиву даних. Оголошення, ініціалізація і виведення масивів. Типові алгоритми опрацювання масивів. Визначення максимального і мінімального елементів та їхніх номерів.	4
10	Тема 10. Обробка одновимірних масивів. Алгоритм упорядкування елементів одновимірного масиву. Інші методи впорядкування масивів.	4
11	Тема 11. Складні типи даних в C++. Двовимірні масиви. Ініціалізація двовимірних масивів. Типові задачі, в яких використовуються двовимірні масиви. Масиви структур. Показники на структури. Передача по посиланню членів масивів структур. Об'єднання й операції з ними.	4
12	Тема 12. Обробка двовимірних масивів. Алгоритми обробки елементів двовимірного масиву.	4
13	Тема 13. Функції. Вказівники та посилання. Поняття про структурне програмування. Оголошення та виклик функцій. Передача аргументів. Прототипи функцій. Область видимості. Локальні і глобальні змінні. Аргументи за замовчуванням. Перенавантаження та шаблони функцій. Рекурсія. Загальний огляд. Разіменування вказівників. Арифметика вказівників. Вказівники. на вказівники. Вказівники. на функції. Посилання. Передача параметрів за посиланням і за значенням. Використання вказівників і посилань із ключовим словом const.	4
14	Тема 14. Обробка рядків Огляд класу string. Виконання базових операцій над об'єктами string. Робота з об'єктами string за допомогою ітераторів.	6
15	Тема 15. Робота з контейнерами std: vector, list. Виконання базових операцій над об'єктами vector, list. Робота з об'єктами vector, list за допомогою ітераторів.	4

16	Тема 16. Робота з контейнерами <code>std: deque, stack, queue</code> . Виконання базових операцій над об'єктами <code>deque, stack, queue</code> . Робота з об'єктами <code>deque, stack, queue</code> за допомогою ітераторів.	6
17	Тема 17. Робота з контейнерами <code>std:map, multimap</code> . Виконання базових операцій над об'єктами <code>map, multimap</code> . Робота з об'єктами <code>map, multimap</code> за допомогою ітераторів.	4
18	Тема 18. Файловий ввід/вивід. Робота з <code>eof</code> . Огляд вводу/виводу. Параметри й аргументи функцій. Аргументи за замовчуванням. Простір імен. Передача аргументів. Прототипи функцій. Область видимості. Перенавантаження та шаблони функцій. Рекурсія. Необхідні кроки роботи з <code>eof</code> . Варіанти та альтернативи.	4

6. САМОСТІЙНА РОБОТА

№ з/п	Назва теми (короткий зміст)	Кількість годин
1	Тема 1. Вступ. Основні поняття та означення Класифікація і особливостями сучасних мов програмування; середовища програмування та елементами вікна середовища програмування Microsoft Visual Studio;	1
2	Тема 2. Синтаксис та семантика мови C++ Множина символів C++ та множина представимих символів. Правила формування констант, ідентифікаторів. Ключові слова. Використання коментарів у програмах. Поняття лексеми.	1
3	Тема 3. Структура програми на мові C++. Оголошення змінних. Склад програми на мові C++. Вихідні файли програми. Виконання програм. Особливості функції <code>main()</code> . Поняття "час життя" і "область видимості". Базові типи даних. Правила приведення базових типів даних. Модифікатори змінних. Автоматичні змінні. Реєстрові змінні. Зовнішні змінні та функції статичні змінні.	1
4	Тема 4. Керування потоками вводу-виводу. Вирази. Основні оператори вводу та виводу на мові C++, зокрема на прикладі <code>printf()</code> та <code>scanf()</code> . Формат вводу-виводу різних типів даних (чисел, рядків, вказівників тощо). Арифметичні операції. Оператор присвоєння . Поняття виразу. Оператори інкремента і декремента. Оператор <code>sizeof</code> . Порозрядні логічні операції. Операції зсуву вліво і вправо. Оператори порівняння. Операція "кома". Пріоритет і порядок виконання операцій.	1
5	Тема 5. Засоби мови C++ для реалізації основних структур алгоритмів. Оператор <code>if</code> . Умовні оператори. Оператори <code>if</code> . Оператори <code>if-else</code> . Умовний оператор <code>?:</code> . Оператор <code>switch</code> .	1
6	Тема 6. Оператори циклу. Цикл <code>while</code> .	1

	Поняття циклу: цикли while, do while, for. Управляючі оператори в циклах: оператор break, оператор continue.	
7	Тема 7. Цикли do-while. Вкладені цикли. Інтегрованій налагоджувач Microsoft Visual Studio.	1
8	Тема 8. Цикл for. Засоби опрацювання елементів циклу for.	1
9	Тема 9. Складні типи даних в C++. Одновимірні масиви. Поняття масиву даних. Оголошення, ініціалізація і виведення масивів. Типові алгоритми опрацювання масивів. Визначення максимального і мінімального елементів та їхніх номерів.	1
10	Тема 10. Обробка одновимірних масивів. Алгоритм упорядкування елементів одновимірного масиву. Інші методи впорядкування масивів.	1
11	Тема 11. Складні типи даних в C++. Двовимірні масиви. Ініціалізація двовимірних масивів. Типові задачі, в яких використовуються двовимірні масиви. Масиви структур. Показчики на структури. Передача по посиланню членів масивів структур. Об'єднання й операції з ними.	1
12	Тема 12. Обробка двовимірних масивів. Алгоритми обробки елементів двовимірного масиву.	1
13	Тема 13. Функції. Вказівники та посилання. Поняття про структурне програмування. Оголошення та виклик функцій. Передача аргументів. Прототипи функцій. Область видимості. Локальні і глобальні змінні. Аргументи за замовчуванням. Перенавантаження та шаблони функцій. Рекурсія. Загальний огляд. Разіменування вказівників. Арифметика вказівників. Вказівники. на вказівники. Вказівники. на функції. Посилання. Передача параметрів за посиланням і за значенням. Використання вказівників і посилань із ключовим словом const.	2
14	Тема 14. Обробка рядків Огляд класу string. Виконання базових операцій над об'єктами string. Робота з об'єктами string за допомогою ітераторів.	2
15	Тема 15. Робота з контейнерами std: vector, list. Виконання базових операцій над об'єктами vector, list. Робота з об'єктами vector, list за допомогою ітераторів.	2
16	Тема 16. Робота з контейнерами std: deque, stack, queue. Виконання базових операцій над об'єктами deque, stack, queue. Робота з об'єктами deque, stack, queue за допомогою ітераторів.	2
17	Тема 17. Робота з контейнерами std:map, multimap. Виконання базових операцій над об'єктами map, multimap. Робота з об'єктами map, multimap за допомогою ітераторів.	2
18	Тема 18. Файловий ввід/вивід. Робота з eof. Огляд вводу/виводу. Параметри й аргументи функцій. Аргументи за замовчуванням. Простір імен. Передача	2

	аргументів. Прототипи функцій. Область видимості. Перенавантаження та шаблони функцій. Рекурсія. Необхідні кроки роботи з eof . Варіанти та альтернативи.	
--	---	--

7. МЕТОДИ НАВЧАННЯ

При викладанні навчальної дисципліни «Основи програмування» застосовуються інформаційні та практичні методи навчання: класичні лекції та лабораторно-практичні заняття з використанням симуляційних лабораторних практикумів, а також консультації з виконання самостійної роботи студентів, письмові завдання.

Методи навчально-пізнавальної діяльності: пояснювально-ілюстративний метод, репродуктивний метод, метод проблемного викладу, частково-пошуковий або евристичний метод, дослідницький метод.

Методи стимулювання й мотивації навчально-пізнавальної діяльності: індуктивні і дедуктивні методи навчання, методи стимулювання і мотивації навчання.

8. МЕТОДИ КОНТРОЛЮ

Відповідно до плану вивчення дисципліни «Основи програмування» передбачається проведення поточного та підсумкового контролю.

Поточний контроль – оцінювання рівня знань, умінь та навичок осіб, які навчаються, що здійснюється в ході навчального процесу шляхом проведення письмового опитування по закінченню розділів (модульний колоквиум). Підсумковий контроль проводиться у формі екзамену.

9. ФОРМА ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ НАВЧАННЯ

Формою підсумкового контролю є **екзамен**, який складається очно (при особливій ситуації – у формі комп'ютерного тесту) в період призначений деканатом або за індивідуальним графіком, який затверджується навчальним планом.

10. СХЕМА НАРАХУВАННЯ ТА РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ СТУДЕНТИ

Накопичення балів протягом семестру

№ з/п	Вид діяльності	Кількість балів за дидактичну одиницю	Кількість	Загальна кількість балів
1	Виконання тестів	2	8	16
2	Виконання лабораторних робіт	4	8	32
3	Виконання завдань СРС	1,5	8	12

Максимальна оцінка	60
--------------------	----

Загальна оцінка знань студентів за поточним контролем

Результати поточного контролю знань студентів в цілому оцінюються в діапазоні від **0** до **60** балів.

Студент допускається до підсумкового контролю за умови виконання вимог навчальної програми та у разі, якщо за поточну навчальну діяльність він набрав не менше **36** балів.

Підсумкове оцінювання знань студентів

Підсумкове оцінювання знань студентів проводиться у формі **екзамену**.

Розподіл балів оцінювання при підсумковому контролі з навчальної дисципліни

Оцінка в балах за поточне оцінювання	Оцінка за національною шкалою
35-40	Відмінно
21-34	Добре
10-20	Задовільно
менше 10	Незадовільно

Під час оцінювання відповіді на окреме питання під час екзамену додатково враховуються допущені недоліки та помилки, якими вважаються:

- неохайне оформлення роботи (не загальноприйняті скорочення, незрозумілий почерк, використання олівців замість чітких чорнил) (мінус **2** бали);

- неточності в назвах окремих термінів та понять (мінус **4** бали).

Критерії оцінювання відповіді на теоретичні питання:

1. Повна відповідь на питання, яка оцінюється **«відмінно»**, повинна відповідати таким вимогам:

- розгорнутий, вичерпний виклад змісту даної у питанні проблеми;
- повний перелік необхідних для розкриття змісту питання фізичних термінів та законів;
- здатність здійснювати порівняльний аналіз різних теорій, концепцій, підходів та самостійно робити логічні висновки й узагальнення;
- уміння користуватись методами наукового аналізу фізичних явищ, процесів і характеризувати їхні риси та форми виявлення;
- демонстрація здатності висловлення та аргументування власного ставлення до альтернативних поглядів на дане питання;

2. Відповідь на питання оцінюється **«добре»**, якщо:

- відносно відповіді на найвищий бал не зроблено розкриття хоча б одного з пунктів, вказаних вище (якщо він явно потрібний для вичерпного розкриття питання) або, якщо:

– при розкритті змісту питання в цілому правильно за зазначеними вимогами зроблені окремі помилки під час: використання формул.

3. Відповідь на питання оцінюється **«задовільно»**, якщо:

– відносно відповіді на найвищий бал не зроблено розкриття чотирьох чи більше пунктів, зазначених у вимогах до нього (якщо вони явно потрібні для вичерпного розкриття питання);

– одночасно присутні чотири чи більше типів недоліків, які окремо характеризують критерій оцінки питання;

– висновки, зроблені під час відповіді, не відповідають правильним чи загально визначеним при відсутності доказів супротивного аргументами, зазначеними у відповіді;

– характер відповіді дає підставу стверджувати, що особа, яка складає іспит, не зовсім правильно зрозуміла зміст питання чи не знає правильної відповіді і тому не відповіла на нього по суті, допустивши грубі помилки у змісті відповіді.

Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту (роботи), практики	для заліку
90-100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
66-73	D	задовільно	
60-65	E		
30-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-29	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

Загальна підсумкова оцінка в балах, за національною шкалою та за шкалою ECTS заноситься до заліково-екзаменаційної відомості, навчальної картки та залікової книжки студента.

11. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

- робоча навчальна програма дисципліни;
- електронний курс на платформі дистанційного навчання;
- плани лекцій, практичних занять та самостійної роботи студентів;
- тези лекцій з дисципліни;
- методичні вказівки до лабораторно-практичних занять для студентів;
- методичні матеріали, що забезпечують самостійну роботу студентів;

- тестові завдання до тем лекцій;
- перелік питань до екзамену.

12. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна:

1. Пекарський Б.Г. Основи програмування: Навчальний посібник. Кондор, 2018.364 с.
2. Васильєв О.Н. Самоучитель С++ з задачами та прикладами (+ віртуальний CD). Наука і техніка, 2016.480 с.
3. Саттер Г. Вирішення складних задач на С++. Вільямс, 2015.400 с.
4. Джордж Хайнеман, Гері Полліс, Стенлі Селков. Алгоритми. Довідник з прикладами на С, С ++, Java і Pythonю.- Діалектика, 2017. 432 с.
5. Основи програмування: навч.посібник / М.Ф.Бондаренко, О.Г.Качко. Х.: Компанія СМІТ, 2008. 432с.
6. Габрусев В. Ю. Лапінський В.В., Нестеренко О.В. Основи операційних систем: Ядро, процес, потік. Тернопіль: Богдан, 2007. 94 с.

Додаткова:

1. С++ Crash Course: A Fast-Paced Introduction./ Lospinoso Josh. ISBN 1593278885. - 2019.- 792с.
2. International Standard ISO/IEC 14882:2014(E) – Programming Language C++, ISBN-13: 978- 0321563842: [Електронний ресурс]. – Режим доступу: <https://isocpp.org/std/the-standard>.
3. C/C++ language and standard libraries reference: [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/en-us/library/hh875057.aspx>. Основи програмування: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія» / В. Г. Зайцев, І. П. Дробязко; КПІ ім. Ігоря Сікорського. 2019. – 240 с. [Електронний ресурс]
4. Основи програмування: навчальний посібник. [за ред. В. М. Рудницького] / І. М. Федотова-Півень, І. В. Миронець, О. Б. Півень, С. В. Сисоєнко, Т. В. Миронюк; Черкаський державний технологічний університет. – Харків : ТОВ «ДІСА ПЛЮС», 2019. – 216 с.

Інформаційні ресурси

1. <https://www.microsoft.com/uk-ua/>
2. <https://stud.com.ua/informatika/>
3. <https://dou.ua/>
4. <http://it.ridne.net/>
5. <https://www.kernel.org/>