

Рівненський державний гуманітарний університет
Кафедра інформаційних технологій та моделювання

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ПРОГРАМУВАННЯ**

Спеціальність **121 Інженерія програмного забезпечення**
Освітня програма «**Інженерія програмного забезпечення**»
Рівень вищої освіти **перший (бакалаврський)**
Факультет **математики та інформатики**

2024 – 2025 навчальний рік

Робоча програма навчальної дисципліни «Програмування» для здобувачів вищої освіти першого (бакалаврського) рівня зі спеціальності 121 Інженерія програмного забезпечення за освітньою програмою «Інженерія програмного забезпечення».

Мова навчання: українська.

Розробник: Шевцова Н.В., доцент кафедри інформаційних технологій та моделювання, кандидат технічних наук.

Робоча програма затверджена на засіданні кафедри інформаційних технологій та моделювання.

Протокол від 27 серпня 2024 року № 8.

Завідувач кафедри



Мороз І. П.

Робочу програму схвалено навчально-методичною комісією факультету математики та інформатики.

Протокол від 3 вересня 2024 року № 7.

Голова навчально-методичної комісії



Гнедко Н. М.

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни			
		денна форма навчання		заочна форма навчання	
Кількість кредитів : 10 Модулів : 2 Змістових модулів : 4 Індивідуальне науково-дослідна робота – Загальна кількість годин : 300 Тижневих годин для денної форми навчання: аудиторних – 4 ; самостійної роботи – 8	Галузь знань: 12 Інформаційні технології Спеціальність: 121 Інженерія програмного забезпечення Освітня програма: «Інженерія програмного забезпечення» Рівень вищої освіти: перший (бакалаврський)	Обов'язкова			
		Рік підготовки			
		1-й			
		Семестр			
		1-й	2-й	1-й	2-й
		Лекції			
		26 год.	24 год.	8 год.	8 год.
		Практичні, семінарські			
		-	-	-	-
		Лабораторні			
		24 год.	26 год.	8 год.	8 год.
		Самостійна робота			
		100 год.	100 год.	134 год.	134 год.
		Індивідуальні завдання:			
-		-			
Вид контролю:					
залік	екзамен	залік	екзамен		
Передумови для вивчення дисципліни: «Дискретний аналіз», «Програмне забезпечення обчислювальних систем», «Іноземна мова (за професійним спрямуванням)»					

2. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Навчальна дисципліна «Програмування» належить до обов'язкових дисциплін професійної підготовки бакалаврів спеціальності 121 Інженерія програмного забезпечення. На ній базується вивчення наступних навчальних дисциплін: «Алгоритми і структури даних», «Об'єктно-орієнтоване програмування», «Основи інженерії програмного забезпечення», «Веб-програмування», «Моделювання та проектування програмного забезпечення».

Метою навчальної дисципліни «Програмування» є формування у майбутніх спеціалістів знань та вмінь, необхідних для розв'язання прикладних задач у ІТ-сфері, ознайомлення з базовими поняттями у структурному програмуванні, оволодіння вміннями розробки алгоритмів та їх реалізації засобами процедурного програмування.

Завданням вивчення дисципліни «Програмування» є отримання знань про процедурний підхід у програмуванні, структурування даних та способи їх опрацювання, набуття та розвиток практичних навичок розв'язування навчальних і практичних задач програмування.

Згідно з освітньо-професійною програмою навчальна дисципліна «Програмування» має забезпечити формування у здобувачів вищої освіти відповідних компетентностей.

Загальні компетентності

K01. Здатність до абстрактного мислення, аналізу та синтезу.

K02. Здатність застосовувати знання у практичних ситуаціях.

K05. Здатність вчитися і оволодівати сучасними знаннями.

Фахові компетентності

K21. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.

K23. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

K26. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

K27. Здатність до алгоритмічного та логічного мислення.

K29. Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.

3. ОЧІКУВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

У результаті освоєння повного курсу навчальної дисципліни «Програмування» у студентів формуються глибокі, міцні і системні знання, які передбачають вільне володіння понятійним апаратом, розуміння основних задач предмету, його мети та завдання.

Згідно з вимогами освітньо-професійної програми здобувачі вищої освіти повинні набути таких **програмних результатів навчання**:

ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПР15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

ПР26. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інженерії програмного забезпечення.

Здобувачі вищої освіти повинні

знати:

- означення алгоритму, його властивостей та способів подання;
- означення програми та етапів підготовки програми до виконання;
- основні концепції алгоритмічних мов;
- базові засоби програмування C/C++, особливості процедурного програмування на базі C/C++;
- способи структурування даних та алгоритми їх обробки.

вміти:

- будувати ефективні алгоритми розв'язку задач;
- здійснювати програмну реалізацію алгоритмів мовами C та C++;
- проводити структурування даних та їх обробку;
- виконувати тестування та налагодження програм.

4. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Модуль 1

Змістовий модуль 1. «Основи програмування алгоритмічною мовою»

Тема 1. Програма. Мови програмування. Основні етапи підготовки програми до виконання. Етапи трансляції. Компілятори та інтерпретатори, компонувальники, бібліотеки, налагоджувачі.

Тема 2. Етапи обробки даних та структура засобів програмування. Структурне програмування. Команди та дані. Структури керування. Структурні мови програмування.

Тема 3. Алгоритм як засіб вирішення задачі. Побудова алгоритму як перший етап вирішення задачі. Визначення та властивості алгоритму. Способи подання алгоритму. Блок-схема як форма опису алгоритму. Методи побудови алгоритму, алгоритмічна декомпозиція.

Тема 4. Мова програмування C++. Основні елементи мови C++. Алфавіт мови. Ідентифікатори, коментарі. Структура програми. Стандартні типи даних. Змінні. Константи. Літерали. Вирази. Операції. Перетворення типів (явне та неявне). Засоби вводу-виводу C++.

Тема 5. Оператори: порожній, складений, переходу, розгалуження, множинного вибору, циклів.

Змістовий модуль 2. «Функції. Макроси»

Тема 6. Складені типи даних мов C/C++. Масиви. Одновимірні та багатовимірні масиви.

Тема 7. Функції. Визначення, виклик, прототип. Механізм передачі параметрів. Глобальні та локальні змінні. Класи пам'яті. Методи розподілення пам'яті статичної, автоматичної та динамічної. Перевантаження функцій. Параметри функцій по замовчуванню. Вбудовані (inline) функції. Передача масивів у функції.

Тема 8 Шаблони функцій. Рекурсивні функції.

Тема 9. Директиви препроцесору. Макроси. Вбудовані функції у порівнянні з макросами.

Модуль 2

Змістовий модуль 3. «Вказівники. Посилання. Динамічний розподіл пам'яті»

Тема 10. Вказівники. Операції над вказівниками. Вказівники як аргументи та результати функцій. Посилання. Передача параметрів за посиланням та за значенням. Масиви та вказівники.

Тема 11. Статичний та динамічний розподіл пам'яті. Створення та знищення динамічних змінних. Динамічні масиви. Вказівники на вказівники. Масиви вказівників. Динамічні багатовимірні масиви.

Тема 12. Вказівники на функції. Масиви вказівників на функції. Функції з довільним числом параметрів. Параметри функції main().

Змістовий модуль 4. «Складені типи. Робота з файлами»

Тема 13. Рядки символів у C/C++. Оголошення рядків. Доступ до елементів рядків. Функції для роботи з рядками.

Тема 14. Структури і об'єднання. Структури з бітовими полями. Об'єднання. Масиви структур.

Тема 15. Обробка текстових файлів. Адресація потоків вводу і виводу. Відкривання файлів для запису, читання, модифікації, до запису в кінець. Функції для запису у файл і читання з файлу.

5. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Назви змістовних модулів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		лек	пр	лаб	інд	с.р.		лек	пр	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
I семестр												
Модуль 1												
<i>Змістовий модуль 1. «Основи програмування алгоритмічною мовою»</i>												
Тема 1. Програма. Мови програмування. Основні етапи підготовки програми до виконання.	10	2		2		6	10	1		1		8
Тема 2. Етапи обробки даних та структура засобів програмування.	10	2				8	10					10
Тема 3. Алгоритм як засіб вирішення задачі.	13	2		1		10	13	1		1		11
Тема 4. Мова програмування C++. Основні елементи мови C++.	15	4		1		10	15	1		1		13
Тема 5. Оператори мови C++.	32	4		8		20	32	1		1		30
Разом за змістовим модулем 1	80	14		12		54	80	4		4		72
<i>Змістовий модуль 2. «Функції. Макроси»</i>												
Тема 6. Складені типи даних мов C/C++. Масиви.	18	4		4		10	18	1		1		16
Тема 7. Функції. Визначення, виклик, прототип.	28	4		4		20	28	1		1		26
Тема 8 Шаблони функцій. Рекурсивні функції.	12	2		2		8	12	1		1		10
Тема 9. Директиви препроцесору. Макроси.	12	2		2		8	12	1		1		10
Разом за змістовим модулем 2	70	12		12		46	70	4		4		62
Разом за модулем 1	150	26		24		100	150	8		8		134
Разом за I семестр	150	26		24		100	150	8		8		134
II семестр												
Модуль 2												
<i>Змістовий модуль 3. «Вказівники. Посилання. Динамічний розподіл пам'яті»</i>												
Тема 10. Вказівники. Операції над	22	4		4		14	22	1		2		19

вказівниками.											
Тема 11. Статичний та динамічний розподіл пам'яті.	30	6	4	20	30	2	1			27	
Тема 12. Вказівники на функції.	20	2	4	14	20	1	1			18	
Разом за змістовим модулем 3	72	12	12	48	72	4	4			64	
Змістовий модуль 4. «Складені типи. Робота з файлами»											
Тема 13. Рядки символів у C/C++.	24	4	4	16	24	1	2			21	
Тема 14. Структури і об'єднання.	30	4	6	20	30	2	1			27	
Тема 15. Обробка текстових файлів.	24	4	4	16	24	1	1			22	
Разом за змістовим модулем 4	78	12	14	52	78	4	4			70	
Разом за модулем 2	150	24	26	100	150	8	8			134	
Разом за II семестр	150	24	26	100	150	8	8			134	
Усього годин	300	50	50	200	300	16	16			268	

6. ТЕМИ СЕМІНАРСЬКИХ ЗАНЯТЬ

Не передбачено навчальним планом

7. ТЕМИ ПРАКТИЧНИХ ЗАНЯТЬ

Не передбачено навчальним планом

8. ТЕМИ ЛАБОРАТОРНИХ ЗАНЯТЬ

№ з/п	Назва теми	Кількість годин	
		денна форма	заочна форма
Семестр I			
1	Структура C++-програми. Вивід та ввід у C++. Змінні та константи.	2	1
2	Типи даних. Операції C++. Вирази. Перетворення типів.	2	
3	Оператори. Оператори розгалуження. Операція умовного виразу.	2	1
4	Оператор множинного вибору. Перелічувальний тип.	2	
5	Програмування циклічних алгоритмів. Оператори goto, while, do while. Програмування циклічних алгоритмів. Обробка числових та символьних послідовностей.	2	1
6	Програмування циклічних алгоритмів. Випадкові числа. Оператор циклу for. Вкладені цикли.	2	1
7	Одновимірні масиви.	2	1
8	Двовимірні масиви.	2	

9	Функції. Визначення. Прототип. Передача параметрів. Параметри по замовчуванню. Глобальні та локальні змінні. Класи пам'яті.	2	1
10	Передача масивів у функції. Перевантаження функцій. Вбудовані (inline) функції.	2	1
11	Шаблони функцій. Рекурсивні функції.	2	1
12	Директиви препроцесору. Макроси.	2	
Разом за I семестр		24	8
Семестр II			
16	Вказівники. Операції над вказівниками. Масиви та вказівники.	2	1
17	Посилання. Передача у функції параметрів за значенням та за посиланням.	2	
18	Створення та знищення динамічних змінних. Динамічні масиви.	2	2
19	Вказівник на вказівник. Масиви вказівників. Динамічні багатовимірні масиви.	2	
20	Вказівники на функції. Масиви вказівників на функції.	2	1
21	Функції з довільним числом параметрів. Параметри функції main().	2	
22	Рядки у C. Робота з рядками. Використання бібліотеки C.	2	2
23	Рядки у C++ (string). Робота з рядками.	2	
24	Структури. Структури з бітовими полями. Вкладені структури.	2	1
25	Масиви структур.	2	
26	Об'єднання.	2	
27	Робота з текстовими файлами. Типи файлових потоків. Створення текстового файлу. Запис у файл. Читання із файлу.	4	1
Разом за II семестр		26	8
Разом		50	16

9. САМОСТІЙНА РОБОТА

№ з/п	Назва теми	Кількість годин	
		денна форма	заочна форма
Семестр I			
1	Програма. Мова програмування. Основні етапи підготовки програми до виконання. Етапи трансляції. Компілятори та інтерпретатори, компонувальники, бібліотеки, налагоджувачі.	6	8
2	Етапи обробки даних та структура засобів програмування. Структурне програмування. Команди та дані. Структури керування. Структурні мови програмування.	8	10
3	Алгоритм як засіб вирішення задачі. Побудова алгоритму як перший етап вирішення задачі. Визначення та властивості алгоритму. Способи подання алгоритму. Блок-схема як форма опису алгоритму. Методи побудови алгоритму, алгоритмічна	10	11

	декомпозиція.		
4	Мова програмування C++. Елементи мови C++. Алфавіт мови. Ідентифікатори, коментарі. Структура програми. Стандартні типи даних. Змінні. Константи. Літерали. Вирази. Операції. Перетворення типів (явне та неявне). Засоби вводу-виводу C++.	10	13
5	Оператори: порожній, складений, переходу, розгалуження, множинного вибору, циклів.	20	30
6	Складені типи даних мов C /C++. Масиви. Одновимірні та багатовимірні масиви.	10	16
7	Функції. Визначення, виклик, прототип. Механізм передачі параметрів. Глобальні та локальні змінні. Класи пам'яті. Методи розподілення пам'яті статичної, автоматичної та динамічної. Перевантаження функцій. Параметри функцій по замовчуванню. Вбудовані (inline) функції. Передача масивів у функції.	20	26
8	Шаблони функцій. Рекурсивні функції.	8	10
9	Директиви препроцесору. Макроси. Вбудовані функції у порівнянні з макросами.	8	10
Разом за I семестр		100	134
Семестр II			
10	Вказівники. Операції над вказівниками. Вказівники як аргументи та результати функцій. Посилання. Передача параметрів за посиланням та за значенням. Масиви та вказівники.	14	19
11	Статичний та динамічний розподіл пам'яті. Створення та знищення динамічних змінних. Динамічні масиви. Вказівники на вказівники. Масиви вказівників. Динамічні багатовимірні масиви.	20	27
12	Вказівники на функції. Масиви вказівників на функції. Функції з довільним числом параметрів. Параметри функції main().	14	18
13	Рядки символів у C/C++. Оголошення рядків. Доступ до елементів рядків. Функції для роботи з рядками.	16	21
14	Структури і об'єднання. Структури з бітовими полями. Об'єднання. Масиви структур.	20	27
15	Обробка текстових файлів. Адресація потоків вводу і виводу. Відкривання файлів для запису, читання, модифікації, до запису в кінець. Функції для запису у файл і читання з файлу.	16	22
Разом за II семестр		100	134
Разом		200	268

Самостійна робота студентів над теоретичним та практичним матеріалом навчальної дисципліни здійснюється в таких формах:

- вивчення теоретичного матеріалу, що викладений на лекційних заняттях та призначеного для самостійного опрацювання;
- індивідуальне виконання навчальних завдань, розв'язування алгоритмічних задач та завдань по розробці алгоритмів та програмуванню.

10. ІНДИВІДУАЛЬНІ ЗАВДАННЯ

Не передбачено

11. МЕТОДИ НАВЧАННЯ

МН1 – словесний метод (лекція, дискусія, співбесіда тощо);

МН2 – практичний метод (практичні заняття);

МН3 – наочний метод (метод ілюстрацій і метод демонстрацій);

МН4 – робота з навчально-методичною літературою (конспектування, тезування, складання реферату);

МН5 – інтерактивний метод (із застосуванням аудіо, відео, новітніх інформаційних технологій та комп'ютерних засобів навчання);

МН6 – самостійна робота (розв'язання завдань).

12. МЕТОДИ ОЦІНЮВАННЯ

МО1 – екзамен;

МО2 – усне або письмове опитування;

МО7 – презентації результатів виконаних завдань та досліджень;

МО9 – захист лабораторних робіт;

МО10 – залік.

13. ЗАСОБИ ДІАГНОСТИКИ РЕЗУЛЬТАТІВ НАВЧАННЯ

- залік (I семестр);
- екзамен (II семестр);
- модульні контрольні роботи;
- презентації результатів виконаних завдань.

Види та методи навчання і оцінювання

Код компетентності	Назва компетентності	Код ПРН	Назва програмного результату навчання	Методи навчання	Методи оцінювання результатів навчання
K01	Здатність до абстрактного мислення, аналізу та синтезу.	ПР07	Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
		ПР13	Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
K02	Здатність застосовувати знання у практичних ситуаціях.	ПР07	Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
		ПР13	Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
K05	Здатність вчитися і оволодівати сучасними	ПР01	Аналізувати, цілеспрямовано шукати і вибирати необхідні	МН1, МН2, МН3, МН4,	МО1, МО2, МО7, МО9,

	знаннями.		для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	MH5, MH6	MO10
K21	Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.	ПР01	Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
		ПР07	Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
		ПР26	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інженерії програмного забезпечення.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
K23	Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.	ПР01	Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
		ПР15	Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
K26	Здатність обгрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.	ПР15	Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10
		ПР26	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інженерії програмного	MH1, MH2, MH3, MH4, MH5, MH6	MO1, MO2, MO7, MO9, MO10

			забезпечення.		
K27	Здатність до алгоритмічного та логічного мислення.	ПР13	Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
		ПР26	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інженерії програмного забезпечення.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
K29	Здатність забезпечити організацію обчислювальних процесів в інформаційних системах різного призначення з урахуванням архітектури, конфігурування, показників результативності функціонування операційних систем і системного програмного забезпечення.	ПР13	Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10
		ПР26	Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі інженерії програмного забезпечення.	МН1, МН2, МН3, МН4, МН5, МН6	МО1, МО2, МО7, МО9, МО10

14. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Результат освітньої діяльності здобувача вищої освіти оцінюється згідно Положення про оцінювання знань і умінь здобувачів вищої освіти РДГУ за такими критеріями оцінювання та рівнями компетентності:

Суми балів за 100-бальною шкалою	Оцінка в ЄКТС	Значення оцінки ЄКТС	Критерії оцінювання	Рівень компетентності	Оцінка за національною шкалою	
					залік	екзамен
90-100	A	відмінно	здобувач вищої освіти виявляє особливі творчі здібності, вміє самостійно здобувати знання, без допомоги викладача знаходить і опрацьовує необхідну інформацію, вміє використовувати набуті знання і вміння для прийняття рішень у нестандартних ситуаціях, переконливо аргументує відповіді, самостійно розкриває власні здібності	високий (творчий)	зараховано	відмінно

82-89	B	добре	здобувач вищої освіти вільно володіє теоретичним матеріалом, застосовує його на практиці, вільно розв'язує вправи і задачі у стандартних ситуаціях, самостійно виправляє допущені помилки, кількість яких незначна	достатній (конструктивно-варіативний)	зараховано	добре
74-81	C	добре	здобувач вищої освіти вмiє зiставляти, узагальнювати, систематизувати iнформацiю пiд керiвництвом викладача, загалом самостiйно застосовувати її на практицi; контролювати власну дiяльнiсть; виправляти помилки, з-помiж яких є суттєвi, добирати аргументи для пiдтвердження думок			
64-73	D	задовiльно	здобувач вищої освіти вiдтворює значну частину теоретичного матерiалу, виявляє знання i розумiння основних положень, за допомогою викладача може аналізувати навчальний матерiал, виправляти помилки, з-помiж яких є значна кiлькiсть суттєвих	середнiй (репродуктивний)	зараховано	задовiльно
60-63	E	задовiльно	здобувач вищої освіти володiє навчальним матерiалом на рiвнi, вищому за початковий, значну частину його вiдтворює на репродуктивному рiвнi			
33-59	FX	незадовiльно з можливистю повторного складання семестрового контролю	здобувач вищої освіти володiє матерiалом на рiвнi окремих фрагментiв, що становлять незначну частину навчального матерiалу	низький (рецептивно-продуктивний)	не зараховано	незадовiльно
1-34	F	незадовiльно з обов'язковим повторним вивченням дисциплiни	здобувач вищої освіти володiє матерiалом на рiвнi елементарного розпiзнання i вiдтворення окремих фактiв, елементiв, об'єктiв			

Пiдсумкова (загальна) оцiнка з навчальної дисциплiни є сумою рейтингових оцiнок (балiв), одержаних за окремі оцiнюванi форми навчальної дiяльностi: поточне та пiдсумкове

оцінювання рівня засвоєння теоретичного матеріалу під час аудиторних занять та самостійної роботи (модульний контроль); оцінки (бали) за практичну діяльність та оцінка за екзамен.

15. РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ ЗДОБУВАЧІ ВИЩОЇ ОСВІТИ

I семестр (залік)

Поточне тестування та самостійна робота									Сума
Змістовий модуль 1					Змістовий модуль 2				
T1	T2	T3	T4	T5	T6	T7	T8	T9	
5	5	5	5	15	10	10	10	5	
Модульний контроль 15					Модульний контроль 15				
50					50				100

II семестр (екзамен)

Поточне тестування та самостійна робота						Екзамен	Сума
Змістовий модуль 3			Змістовий модуль 4				
T10	T11	T12	T13	T14	T15		
5	5	5	5	5	5		
Модульний контроль 15			Модульний контроль 15				
30			30			40	100

T1, T2, ..., T15 – теми змістових модулів.

16. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

В якості навчально-методичного забезпечення використовується основна та допоміжна література з навчальної дисципліни, матеріал лекцій, вказівки для проведення лабораторних робіт, варіанти завдань для самостійної та індивідуальної роботи студентів, варіанти модульних контрольних робіт та теоретичні питання до заліку та екзамену.

Вказівки до виконання лабораторних розміщені:

<https://do.rshu.edu.ua/enrol/index.php?id=164>

17. ПИТАННЯ ДЛЯ ПІДГОТОВКИ ДО ПІДСУМКОВОГО КОНТРОЛЮ

Питання до екзамену

- Програма. Мова програмування. Основні етапи підготовки програми до виконання.
- Етапи трансляції. Компілятори та інтерпретатори. Етапи трансляції. Компонувальники, бібліотеки, налагоджувачі.
- Компілятори. Фази компіляції. Поняття лексеми.
- Структурне програмування. Структурні мови програмування.
- Алгоритм. Визначення та властивості алгоритму. Способи подання алгоритму. Блок-схема.
- Елементи мови C++. Алфавіт. Ідентифікатори, коментарі. Структура програми.
- Ввід-вивід у C++. Форматування вводу-виводу.
- Стандартні типи даних. Змінні. Константи. Літерали.
- Цілочислові типи даних у C++. Робота з цілочисловими типами.
- Оголошення змінних, констант. Літерали. Приклади літералів.
- Вирази. Операції.
- Перелічуваний тип. Використання перелічувального типу. Приклади.
- Арифметичні операції. Операції присвоєння. Пріоритет та асоціативність.
- Логічний тип. Операції відношення та логічні операції. Логічні вирази. Використання логічних виразів.

15. Перетворення типів у C++. Явне та неявне приведення типів. Ієрархія типів.
16. Вирази у мові C++.
17. Оператори переходу у C++.
18. Оператор розгалуження if. Вкладення if.
19. Оператор множинного вибору switch.
20. Оператори циклу у C++.
21. Оператори циклу з передумовою. Оператори циклу з післяумовою. Порівняльна характеристика.
22. Оператори циклу. Оператор for.
23. Вкладені цикли.
24. Обробка послідовності чисел (символів).
25. Випадкові числа. Генерація випадкових чисел у програмі.
26. Бібліотечні функції smath.
27. Обробка одновимірних масивів.
28. Одновимірні масиви. Оголошення та ініціалізація. Розміщення у пам'яті. Індксація.
29. Багатовимірні масиви. Оголошення та ініціалізація. Розміщення у пам'яті. Обробка.
30. Функції у мові C++. Визначення, прототип, виклик.
31. Функції, що не повертають значення. Функції, що повертають значення. Оператор return.
32. Функції у C++. Формальні та фактичні параметри. Механізм передачі параметрів.
33. Глобальні та локальні змінні. Оголошення, область видимості, час існування.
34. Класи пам'яті у C++. Статичний клас пам'яті.
35. Перевантаження функцій у C++. Функції з параметрами по замовчуванню. Неоднозначний виклик перевантаженої функції.
36. Функції у C++. Вбудовані функції.
37. Передача одновимірних масивів у функції у C++.
38. Передача багатовимірних масивів у функції у C++.
39. Шаблони функцій. Явна та неявна конкретизація шаблону.
40. Перевантаження шаблонів функцій. Приклади.
41. Рекурсивні функції.
42. Поняття вказівника. Оголошення та ініціалізація. Мета використання.
43. Вказівний тип у C/C++. Операції над вказівниками.
44. Зв'язок масивів та вказівників. Обробка одновимірних масивів за допомогою вказівників.
45. Вказівники. Вказівники як аргументи функцій та результати функцій.
46. Вказівники та посилання C++. Порівняльна характеристика.
47. Посилання. Посилання як аргументи та результати функцій у C++.
48. Передача параметрів у функції за значенням та посиланням. Порівняльна характеристика.
49. Створення та знищення динамічних змінних, масивів.
50. Створення та обробка динамічних масивів. Видалення та вставка елементів у динамічний масив.
51. Вказівник на вказівник. Масиви вказівників. Динамічні двовимірні масиви.
52. Вказівники на функції. Масиви вказівників на функції.
53. Рядки. Подання рядків у пам'яті. Ініціалізація. Рядкові літерали
54. Рядки. Ввід-вивід рядків.
55. Масиви рядків. Оголошення. Ініціалізація. Обробка.
56. Структуровані типи даних: рядки у C. Порівняння рядків. Конкатенація рядків.
57. Структуровані типи даних: рядки у C. Стандартні функції пошуку у рядку.
58. Бібліотечні функції копіювання, перетворення рядків. Приклади.
59. Бібліотечні функції для об'єднання рядків, розбиття рядка на лексеми.
60. Структури та об'єднання у C++. Мета використання.
61. Структури. Оголошення та ініціалізація. Розміщення у пам'яті. Приклади.
62. Структури. Вкладені структури. Доступ до полів структур, вкладених структур.
63. Структури. Вказівники на структури. Динамічні структури.

64. Структури з бітовими полями. Оголошення та ініціалізація. Обробка.
65. Структури як параметри та результати функцій.
66. Масиви структур. Оголошення, ініціалізація. Обробка.
67. Об'єднання. Розміщення у пам'яті. Ініціалізація. Обробка.
68. Об'єднання як параметри та результати функцій.
69. Динамічні масиви структур. Створення, обробка, знищення. Приклади.
70. Параметри функції main().
71. Ввід-вивід у мові С символів, рядків.
72. Ввід-вивід у мові С даних числових типів.
73. Файли. Текстові, двійкові файли. Порівняльна характеристика.
74. Обробка текстових файлів.
75. Текстові файли. Форматований ввід-вивід.
76. Текстові файли. Запис та зчитування структур, масивів структур.

18. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

Основна:

1. Іванов Є.О., Ліндер Я.М., Жереб К.А. Основи мови програмування C++: навчальний посібник. К.: Логос, 2020. 90 с.
2. Васильєв О. Програмування на C++ в прикладах і задачах: навчальний посібник. К.: Видавництво «Ліра-К», 2017. 382 с.
3. Боровльова С. Ю., Швед А. В. Базовий C++: навчальний посібник. Миколаїв: Вид-во ЧНУ ім. Петра Могили, 2017. 116 с.
4. Вступ до програмування мовою C++. Організація обчислень : навчальний посібник / Белов Ю. А., Карнаух Т. О., Коваль Ю. В., Ставровський А. Б. К.: Видавничо-поліграфічний центр "Київський університет", 2012. 175 с
5. Scott Meyers Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14 1st Edition. O'Reilly, 2014. 334 с.
6. Ivor Horton. Beginning C++20: From Novice to Professional 6th Edition. Apress, 2020. 825 p.
7. Tony Gaddis. Starting Out with C++ from Control Structures to Objects. NY: Pearson, 2017. 1344 p.

Допоміжна:

1. Richard Reese. Understanding and Using C Pointers Core techniques for memory management. O'Reilly, 2014. 226 p.
2. Paul Deitel, Harvey Deitel. C++ How to Program: 10 edition. NY: Pearson, 2016. 1080 p.
3. Michael Main, Walter Savitch. Data Structures and Other Objects Using C++: 4th edition. NY: Pearson, 2013. 848 p.

19. ЕЛЕКТРОННІ ІНФОРМАЦІЙНІ (ІНТЕРНЕТ) РЕСУРСИ ТА ТЕХНІЧНІ ЗАСОБИ НАВЧАННЯ

Електронні інформаційні ресурси:

- The Standard. URL: <https://isocpp.org/std/the-standard>
- C++ language documentation. URL: <https://learn.microsoft.com/en-us/cpp/cpp/?view=msvc-170>
- Уроки програмування на C++. URL: <https://acode.com.ua/uroki-po-cpp/>
- Learn.cpp. URL: <https://www.learncpp.com/>
- C++ Tutorial for Complete Beginners URL: <https://www.udemy.com/course/free-learn-c-tutorial-beginners/>
- C++ Tutorial. URL: <https://www.sololearn.com/Course/CPlusPlus/>

Технічні засоби та прикладне програмне забезпечення:

- ПЕОМ;

- ОС MS Windows*;
- Середовища та компілятори мов програмування : Visual Studio Community, Visual Studio Code.