

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ВОДНОГО ГОСПОДАРСТВА ТА  
ПРИРОДОКОРИСТУВАННЯ**

Навчально-науковий інститут кібернетики, інформаційних технологій та інженерії

**04-04-50S**

<b>СИЛАБУС</b>		<b>SYLLABUS</b>	
Програмування		Programming	
Шифр за ОП	OK 13	Code in Degree Programme	
Освітній рівень: Бакалаврський (перший)		Level of Education: Bachelor's (first)	
Галузь знань Інформаційні технології	12	Field of Knowledge Information Technology	
Спеціальність Комп'ютерна інженерія	123	Field of Study Computer Engineering	
Освітня програма: Комп'ютерна інженерія		Degree Programme: Computer Engineering	

Силабус навчальної дисципліни «Програмування» для здобувачів вищої освіти ступеня «бакалавр», які навчаються за освітньо-професійною програмою «Комп'ютерна інженерія», спеціальності 123 «Комп'ютерна інженерія». Рівне. НУВГП. 2024. 12 стор.

ОП на сайті університету:  
<https://ep3.nuwm.edu.ua/22990/>

Розробник силабусу: Сидор Андрій Іванович, к.т.н., в.о. завідувача кафедри обчислювальної техніки

Силабус схвалений на засіданні кафедри  
Протокол №14 від 14.06.2024 року


В.о. завідувача кафедри:  
Сидор А.І., к.т.н..

Керівник (гарант) ОП: Сидор А.І., к.т.н..

Схвалено науково-методичною радою з якості ННІ КІТІ  
Протокол №7 від 17.06.2024 року

Голова науково-методичної ради з якості ННІ: Мартинюк П.М., д.т.н., професор.  
Попередня версія силабусу: -

ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ	
Програмування	
ЗАГАЛЬНА ІНФОРМАЦІЯ	
Ступінь вищої освіти	бакалавр
Освітня програма	Комп'ютерна інженерія
Спеціальність	123 Комп'ютерна інженерія
Рік навчання, семестр	1-й рік, 1-й семестр
Кількість кредитів	5,5
Лекції:	28/6 годин
Лабораторні заняття:	28/12 годин
Самостійна робота:	109/147 годин
Курсова робота:	Ні
Форма навчання	денна/заочна
Форма підсумкового контролю	екзамен
Мова викладання	державна

ІНФОРМАЦІЯ ПРО РОЗРОБНИКА (ІВ)	
Лектор 	Сидор Андрій Іванович к.т.н., в.о. завідувача кафедри обчислювальної техніки.
Вікіситет	<a href="https://wiki.nuwm.edu.ua/index.php/Сидор_Андрій_Іванович">https://wiki.nuwm.edu.ua/index.php/Сидор_Андрій_Іванович</a>
ORCID	<a href="https://orcid.org/0000-0003-4911-7034">https://orcid.org/0000-0003-4911-7034</a>
Як комунікувати	<a href="mailto:a.i.sydor@nuwm.edu.ua">a.i.sydor@nuwm.edu.ua</a>
ІНФОРМАЦІЯ ПРО НАВЧАЛЬНУ ДИСЦИПЛІНУ	
Мета та завдання	

Мета дисципліни "Програмування" полягає в наданні студентам необхідних знань, навичок та розуміння основ програмування для успішної реалізації програмних проектів та розвитку як професіоналів в галузі інформаційних технологій.

Завдання дисципліни включають:

Навчання основних принципів програмування: структури даних, алгоритми, логіка програмування та парадигми програмування.

Розгляд та розвиток навичок роботи з популярними мовами програмування, такими як C++, а також з іншими інструментами розробки програмного забезпечення.

Вивчення методологій та підходів до розробки програмного забезпечення, включаючи об'єктно-орієнтоване програмування, функціональне програмування та інші.

Практичне застосування отриманих знань у вирішенні завдань та реалізації програмних проектів різної складності.

Розвиток навичок аналізу, проектування та відладки програмного забезпечення.

Засвоєння навичок командної роботи та спільної розробки програмного коду з використанням систем керування версіями та інших інструментів співпраці.

Залучення студентів до вивчення та аналізу сучасних тенденцій у світі програмування та інформаційних технологій.

**Посилання на розміщення освітнього компонента на навчальній платформі Moodle, на платформі освітніх програм та їхніх освітніх компонентів**

<https://exam.nuwm.edu.ua/course/view.php?id=5667>

#### **Передумови вивчення\***

**(місце освітнього компоненту в структурно-логічній схемі)**

Дисципліни, що передують вивченню: дисципліна читається на першому курсі в першому семестрі і не передбачає попереднього вивчення дисциплін.

Отримані навички можуть використовуватись при подальшому вивченні дисциплін: ОК 14 Практична підготовка з програмування

#### **Компетентності**

Z2 Здатність вчитися і оволодівати сучасними знаннями.

Z5. Здатність спілкуватися іноземною мовою.

Z8. Здатність працювати в команді.

P2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

P3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж.

P7. Здатність використовувати та впроваджувати нові технології, включаючи технології розумних, мобільних, зелених і безпечних обчислень, брати участь в модернізації та реконструкції комп'ютерних систем та мереж, різноманітних вбудованих і розподілених додатків, зокрема з метою підвищення їх ефективності.

P8. Готовність брати участь у роботах з впровадження комп'ютерних систем та мереж, введення їх до експлуатації на об'єктах різного призначення.

**Програмні результати навчання (ПРН). Результати навчання (РН)\***

*N5. Мати знання основ економіки та управління проектами.*

*N10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосунків, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.*

*N12. Вміти ефективно працювати як індивідуально, так і у складі команди*

*N13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів.*

*N24. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.*

### Структура та зміст навчальної дисципліни

№	Теми (лекції)	Опис лекції	№	Теми лабораторних занять
<b>МОДУЛЬ 1. Основи програмування</b>				
1	Вступ до програмування (2 год.) <i>N5, N10</i>	Введення в мови програмування C і C++. Введення в розробку програмного забезпечення. Встановлення IDE (Інтегрованого Середовища Розробки). Вирішення найбільш поширених проблем початківців в C++	1	Ввід вивід інформації. (2 год.)
2	Загальна структура програми (2 год.) <i>N5, N13</i>	Налаштування компілятора: Попередження та помилки. Налаштування компілятора: Вибір стандарту мови C++. Структура програм. Коментарі. Змінні, Ініціалізація і Присвоювання. Об'єкти cout, cin і endl. Функції. Параметри і аргументи функцій. Локальна область видимості. Ключові слова та ідентифікатори.	2	Використання математичних операцій. (2 год.)
3	Основи C++ (2 год.) <i>N12, N5</i>	Оператори. Базове форматування коду. Прототип функції і Попереднє оголошення. Багатофайлові програми. Заголовкові файли. Директиви препроцесора. Header guards і директива #pragma once. Конфлікт імен і простір імен std.	3	Використання функцій. (2 год.)
4	Змінні і основні типи даних в C++ (2 год.) <i>N13, N10</i>	Відлагодження програм: степпінг і точки зупину. Ініціалізація, Присвоювання і Оголошення змінних. Тип даних void. Розмір типів даних. Цілочисельні типи даних: short, int і long.	4	Базове форматування коду. (2 год.)

5	Типи даних C++ (2 год.) <i>N12, N24</i>	Фіксований розмір цілочисельних типів даних. Типи даних з плаваючою крапкою: float, double і long double. Логічний тип даних bool. Символьний тип даних char. Літерали і магічні числа. const, constexpr і символічні константи.	5	Робота з декількома функціями. (2 год.)
6	Оператори в C++ (2 год.) <i>N12, N24</i>	Пріоритет операцій і правила асоціативності. Арифметичні оператори. Інкремент, декремент і побічні ефекти. Умовний тернарний оператор, оператор sizeof і Кома. Оператори порівняння. Логічні оператори: !, АБО, НЕ.	6	Використання операторів та різних типів даних. (2 год.)
7	Область видимості в C++ (2 год.) <i>N10, N12</i>	Побітові оператори. Бітові флаги та бітові маски. Блоки стейтментів. Глобальні змінні. Статичні змінні. Зв'язки, область видимості та тривалість життя. Простір імен. using-стейтменти. Неявна конвертація типів даних.	7	Командна розробка простих програм. (2 год.)
<b>МОДУЛЬ 2. Основні конструкції в програмуванні</b>				
8	Інші типи змінних в C++ (2 год.) <i>N10, N24</i>	Явна конвертація типів даних. Введення в std::string. Перерахування (перелічуваний тип даних). Класи enum. Псевдоніми типів: typedef і type alias. Структури. Вивід типів: ключове слово auto. Оператори управління потоком виконання програм.	8	Робота з рядками (2 год.)

9	Цикли і розгалуження в C++ (2 год.) <i>N10, N24</i>	Оператори умовного розгалуження if/else. Оператор switch. Оператор goto. Цикл while. Цикл do while. Цикл for. Оператори break і continue. Генерація рандомних чисел. Обробка некоректного користувачького вводу.	9	Використання циклі та розгалужень (2 год.)
10	Масиви та Рядки, в C++ (2 год.) <i>N10, N24</i>	Масиви. Фіксовані масиви. Масиви і цикли. Двовимірні масиви. Рядки C-style. Клас std::string_view.	10	Робота з масивами (2 год.)
11	Вказівники в C++ (2 год.) <i>N10, N24</i>	Вказівники. Нульові вказівники. Вказівники і масиви. Адресна арифметика і індексація масивів. Символьні константи рядків C-style. Динамічне виділення пам'яті. Динамічні масиви.	11	Робота з динамічними масивами. (2 год.)
12	Посилання в C++ (2 год.) <i>N10, N12</i>	Посилання. Посилання і const. Оператор доступу до членів через вказівник. Цикл foreach. Вказівники на вказівники. Введення в std::array. std::vector (вектори). Введення в літератори. Алгоритми в Стандартній бібліотеці C++.	12	Використання генератора випадкових чисел. (2 год.)
13	Функції в C++. (2 год.) <i>N10, N24</i>	Параметри і аргументи функцій. Передача по значенню. Передача по посиланню. Вбудовані функції.	13	Робота з масивами та функціями. (2 год.)

14	Робота з функціями в C++. (2 год.) N12, N24	Перевантаження функцій. Параметри за замовчуванням. Вказівники на функції. Стек і Купа. Ємність вектора. Рекурсія і Числа Фібоначчі	14	Командна розробка програм з використанням функцій (2 год.)
----	---	---	----	--

### Форми та методи навчання

*Методи навчання: демонстрація, навчальна дискусія, дебати; технології викладання: тренінги, аналіз конкретних ситуацій, обговорення, мультимедійні презентації, міні-лекції, ситуаційні дослідження, навчання на основі досвіду та інші*

### Інструменти, обладнання, програмне забезпечення

*Visual Studio*

### Порядок оцінювання програмних результатів навчання/результатів навчання

- Сума балів = 100:
- 60 – поточна робота;
- 40 – модульний контроль;
- Розподіл балів:
- Модульні контрольні роботи: 40 балів
- 1-й модульний контроль 20 балів, (Рівень1 -18 питань 0.5 бали за питання, Рівень 2 -10 питань 0.6 бали за питання, Рівень 3 5 питань 1 бал за питання)
- 2-й модульний контроль 20 балів, (Рівень1 -18 питань 0.5 бали за питання, Рівень 2 -10 питань 0.6 бали за питання, Рівень 3 5 питань 1 бал за питання)
- Лабораторні роботи: 60 балів, 14 балів за відвідування лекційних завдань 3 бали за лабораторну роботу (1 бал – виконання лабораторної роботи; 1 бал – усний захист лабораторної роботи 1 бал – виконання індивідуального завдання по темі лабораторної.) 4 бали – вчасний захист лабораторних робіт
  - Додаткові бали (участь олімпіадах, конференціях, написання статей, активна студентська діяльність тощо) 5-10 балів.

### Рекомендована література (основна, допоміжна)



**Основна література:**

1. *Williams, Anthony. C++ Concurrency in Action: Practical Multithreading. 2nd ed., Manning Publications, 2019.*
2. *Gregoire, Marc. Professional C++. 5th ed., Wrox, 2021.*
3. *Josuttis, Nicolai M. C++ Move Semantics: The Complete Guide. Nicolai Josuttis, 2019.*
4. *Stroustrup, Bjarne. A Tour of C++. 2nd ed., Addison-Wesley Professional, 2018.*
5. *Koenig, Andrew. C++ Core Guidelines Explained: Best Practices for Modern C++. Addison-Wesley Professional, 2020.*

**Допоміжна література**

1. *Vandevorde, David, Nicolai M. Josuttis, and Douglas Gregor. C++ Templates: The Complete Guide. 2nd ed., Addison-Wesley Professional, 2017.*
2. *Malhotra, M. S. C++ High Performance: Boost and optimize the performance of your C++17 code. 2nd ed., Packt Publishing, 2020.*
3. *Lakos, John. Large-Scale C++ Volume I: Process and Architecture. Addison-Wesley Professional, 2019.*
4. *Van Riper, Kevin, and Paul Drews. Modern C++ Programming with Test-Driven Development: Code Better, Sleep Better. O'Reilly Media, 2019.*
5. *Hedley, John. Data Structures and Algorithm Analysis in C++. 4th ed., Pearson, 2019.*

**Інформаційні ресурси в Інтернеті**

Офіційні сайти та документація:

C++ Standard Documentation

<https://isocpp.org/>

Офіційний сайт з інформацією про стандарт C++, новини, ресурси та посилання на документацію.

Освітні ресурси та курси:

Coursera

<https://www.coursera.org/courses?query=c%2B%2B>

Курси від провідних університетів та компаній, таких як University of Illinois та Google.

**ПОЛІТИКИ ВИКЛАДАННЯ ТА НАВЧАННЯ**

**Перелік соціальних, «м'яких» навичок (soft skills)**

**1. Комунікаційні навички:**

Письмова та усна комунікація - Програмісти часто працюють у команді, спілкуються з менеджерами, замовниками та колегами. Важливо вміти чітко пояснювати свої думки, писати зрозумілий код з коментарями, а також готувати технічну документацію.

**2. Проблемне мислення:**

Аналітичне та критичне мислення - Програмісти мають вміти аналізувати проблеми, розбивати їх на менші частини і знаходити ефективні рішення. Критичне мислення допомагає оцінювати різні варіанти вирішення проблеми та вибрати найоптимальніший.

**3. Управління часом:**

Планування та організація - Вміння ефективно планувати свій робочий час, розподіляти задачі та дотримуватись дедлайнів. Це допомагає зменшити стрес і підвищити продуктивність.

**4. Співпраця та командна робота:**

Командна робота - Програмісти часто працюють у команді над спільними проектами. Вміння співпрацювати, слухати та поважати думки інших є ключовим для успішної командної роботи.

**5. Постійне навчання:**

Навчання та адаптивність - Т-індустрія постійно змінюється, з'являються нові технології та методи. Програмістам необхідно постійно навчатися новим навичкам і бути готовими адаптуватися до змін.

**Дедлайни та перескладання**

Захист лабораторної роботи здійснюється на поточному занятті.

У випадку пропуску заняття можна відпрацювати лабораторну оформивши відповідний звіт.

На здачу кожного з модулів студенту надається одна спроба.

Прийом звітів та їх поточне оцінювання завершується на останньому підсумковому занятті.

У разі успішної здачі двох модулів та лабораторних робіт в загальній сумі більше 60 балів студент претендує на пере зарахування поточної складової балів як підсумкового контролю.

Якщо студент виявив бажання перездати модульні контролю або не набрав відповідної кількості балів він іде на екзамен.

У разі якщо здобувач не набрав достатньої кількості балів після екзамену, його відправляють на комісію.

**Правила академічної доброчесності**

За списування під час проведення модульного контролю чи підсумкового контролю, студент позбавляється подальшого права здавати матеріал і у нього виникає академічна заборгованість.

Здобувачі не допускаються до списування та обману. За порушення принципів академічної доброчесності викладач може накладати такі санкції:

- усне зауваження;
- попередження про можливість притягнення до академічної відповідальності;
- зниження чи анулювання результатів оцінювання навчального завдання здобувача вищої освіти;
- повторне виконання навчального завдання;
- призначення додаткового навчання з питань академічної доброчесності;
- призначення додаткових контрольних заходів (додаткові індивідуальні навчальні завдання, тести тощо).

• Документи стосовно академічної доброчесності (про плагіат, порядок здачі курсових робіт, кодекс честі студентів, документи Національного агентства стосовно доброчесності) наведені на сторінці ЯКІСТЬ ОСВІТИ сайту НУВГП - <http://nuwm.edu.ua/sp/akademichna-dobrochesnistj>

**Вимоги до відвідування**

*• Пропущені заняття відпрацьовуються здобувачами самостійно. Електронні варіанти лекцій доступні на платформі moodle та/або в групі Telegram відповідного курс, лабораторні виконуються у вигляді звіту та захищаються на наступному занятті чи на консультації, індивідуальні завдання надаються для студентів, що знаходяться на індивідуальному плані навчання.*

Автор  
В.О. завідувача кафедри ОТ

Андрій СИДОР

Затверджено

Проректор з науково-педагогічної та навчальної  
роботи

Валерій СОРОКА



документ підписаний КЕП  
Номер документа СИЛ №767  
Підписувач Сорока Валерій Степанович  
Підписувач (дані КЕП):  
Сертифікат 58E2D9E7F900307B04000000807E2D0054327D00