



## Силабус освітнього компонента

Програма навчальної дисципліни



## Основи програмування

### Шифр та назва спеціальності

122 – Комп'ютерні науки

### Інститут

ННІ комп'ютерних наук та інформаційних технологій

### Освітня програма

Комп'ютерні науки. Штучний інтелект та управління проектами

### Кафедра

Системний аналіз та інформаційно-аналітичні технології

### Рівень освіти

Бакалавр

### Тип дисципліни

Спеціальна (фахова), Обов'язкова

### Семестр

1, 2

### Мова викладання

Українська

## Викладачі, розробники



### Любченко Наталія Юріївна

[Nataliia.Liubchenko@khpi.edu.ua](mailto:Nataliia.Liubchenko@khpi.edu.ua)

Кандидат технічних наук, доцент, доцент кафедри системного аналізу та інформаційно-аналітичних технологій

Досвід роботи – 20 років. Автор понад 100 наукових та навчально-методичних праць. Провідний лектор з дисциплін: «Основи програмування», «Технології та засоби великих даних», «Семантичний WEB», «Об'єктно-орієнтоване програмування», «Основи розподілених та паралельних обчислень».

[Детальніше про викладача на сайті кафедри](#)

## Загальна інформація

### Анотація

Завданням дисципліни є набуття студентами теоретичних знань про принципи побудови сучасних мов програмування, вивчення основних конструкцій мов, типових задач алгоритмічного програмування, набуття досвіду роботи в інтегрованому середовищі розробки програм (на прикладі компілятора Visual Studio C++) та проходженні повного циклу розробки (розробка алгоритму - кодування - компіляція - відлагодження - документування). Предметом вивчення навчальної дисципліни є основи алгоритмічного мислення, а також фундаментальні підходи до процедурного та об'єктно-орієнтованого програмування.

### Мета та цілі дисципліни

Метою курсу є ознайомлення з основними середовищами та технологіями програмування, основами створення програмних продуктів, вивчення основних понять процедурного та об'єктно-орієнтованого програмування, а також базових понять алгоритмічного програмування, методів аналізу та реалізації найпростіших алгоритмів засобами мови програмування C++.

## Формат занять

Лекції, лабораторні роботи, практичні заняття, самостійна робота, консультації. Індивідуальне розрахункове завдання, курсова робота, модульні тести. Підсумковий контроль – іспит.

## Компетентності

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

ЗК7. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК8. Здатність генерувати нові ідеї (креативність).

ЗК9. Здатність працювати в команді.

СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

## Результати навчання

РН9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук.

## Обсяг дисципліни

Загальний обсяг дисципліни 360 год. (12 кредитів ECTS): лекції – 64 год., лабораторні роботи – 64 год., практичні заняття – 64 год., самостійна робота – 168 год.

## Передумови вивчення дисципліни (пререквізити)

Загальні знання у галузі інформаційних технологій, загальне володіння сучасним персональним комп'ютером.

## Особливості дисципліни, методи та технології навчання

Лекції проводяться інтерактивно з використанням мультимедійних технологій.

Навчальні матеріали, в тому числі відеозаписи лекцій, доступні студентам через засоби Microsoft Teams. Вивчення курсу потребує використання програмного забезпечення Visual Studio (або інш.), крім загально вживаних програм і операційних систем.

## Програма навчальної дисципліни

### Теми лекційних занять

#### Тема 1. Арифметичні та логічні основи комп'ютера

Поняття про системи числення (позиційну та непозиційну). Перетворення двійкових, вісімкових, десяткових та шістнадцяткових чисел в інші системи числення. Арифметичні дії над двійковими числами.

#### Тема 2. Аналіз алгоритмів програм

Основні алгоритмічні блоки. Алгоритми та програми. Блок-схема як засіб графічного зображення алгоритмів. Лінійні алгоритми. Алгоритми з розгалуженням. Цикли та циклічні структури.

#### Тема 3. Введення в програмування

Види ПЗ. Етапи розробки ПЗ. Структура програми, літерали, синтаксис мови, змінні та константи, сфера їх дії. Інтегроване середовище редагування, налагодження, компіляції та створення програм. Найменування змінних, обмеження на використання певних символів. Типи даних, їх розміри. Ключові слова. Константи. Оголошення змінних.

#### Тема 4. Базові типи даних.

Цілі та дійсні числа. Символьні дані, коди та алгоритми кодування. Логічні дані. Адресні дані та адресна арифметика. Фізичний рівень подання базових типів даних, системи адресації. Поняття типізації у мовах програмування.

#### Тема 5. Введення/виведення в мові C/C++.

Структура програми. Область дії змінних. Перетворення типів. Введення/виведення в мові C. Стандартні функції введення/виведення мов C та C++. Використання коментарів. Директиви препроцесора.

#### Тема 6. Класи пам'яті. Основні операції в C/C++

Загальне розподілення пам'яті. Операції присвоєння. Повна та скорочена форма операції присвоєння. Операції інкремента та декремента. Арифметичні операції та математичні функції мови C/C++. Логічні та побітові операції. Пріоритет виконання операцій.

#### Тема 7. Умовні оператори

Структури управління. Структура вибору if. Структура вибору if/else. Обробка складних умов в мові C/C++. Багатоваріантний вибір switch. Приклади програм з реалізованими структурами управління.

#### Тема 8. Оператори циклу

Три види операторів циклу. Вкладені цикли. Структура повторення for. Блок-схема алгоритму структури for. Синтаксис структури for. Приклади структур for. Структура циклу з передумовою while. Структура циклу з післяумовою do/while. Рекомендації щодо вибору структур повторення в залежності від алгоритму. Приклади програм з реалізованими структурами повторення. Оператори break і continue.

#### Тема 9. Статичні масиви

Статичні одновимірні та багатовимірні масиви. Алгоритми упорядкування масивів. Лінійний пошук, бінарний пошук. Стратегії упорядкування.

#### Тема 10. Робота з одновимірними масивами

Масиви як структурований тип даних. Визначення масивів. Синтаксис об'яви масивів. Поняття "елемент масиву" та "індекс масиву". Синтаксис використання масивів. Алгоритми роботи з послідовностями. Сортування одновимірних масивів. Алгоритми пошуку в одновимірних масивах..

#### Тема 11. Робота з багатовимірними масивами

Приклади програм з багатовимірними масивами. Особливості реалізації двовимірних масивів. Зв'язок між лінійним та багатовимірним індексами. Реалізація алгоритмів, пов'язаних з обробкою матриць. Алгоритми перетворення матриць. Алгоритми заповнення, копіювання, відображення, повороту в двовимірних масивах.

#### Тема 12. Загальні поняття щодо структурного програмування

Програмні модулі мови C. Функції математичної бібліотеки. Функції бібліотеки стандартного введення/виведення. Використання функцій при розробці програм на мові C. Визначення функцій. Прототипи функцій та файли заголовків. Розробка власних функцій. Виклик функції за значенням. Способи обміну інформацією між функціями. Локальні та глобальні змінні.

#### Тема 13. Функції користувача в мові C/C++

Визначення функції. Внутрішні та зовнішні змінні. Значення, що повертає функція. Розробка програми у вигляді проекту. Зовнішні змінні. Використання C-препроцесора. Включення додаткових файлів, макропідстановки, умовна компіляція.

#### Тема 14. Рекурсія

Приклади використання рекурсії. Реалізація стандартних алгоритмів обробки масивів за допомогою рекурсії.

#### Тема 15. Робота зі структурами

Створення власних типів даних за допомогою структур. Об'ява структур. Об'ява структурних змінних. Доступ до елементів структури. Використання масивів структур. Використання структур з функціями.

#### Тема 16. Показчики. Динамічна пам'ять

Моделі пам'яті у C та їх особливості. Модель процесу виконання програми, адресний простір процесу. Типізовані показчики. Операції взяття адреси, розіменування показчика. Динамічні масиви. Властивості динамічних масивів. Показчики на елементи масивів. Адресна арифметика. Масиви як параметри функцій. Кваліфікатор const. Показчики на структури. Використання показчиків як аргументів функцій. Показчик на функцію.

### Тема 17. Поняття С-рядків

Логічна структура, подання, операції, алгоритми, засоби обробки рядків у мові програмування С.

### Тема 18. Робота з рядками в С/С++

С-рядки, особливості масивів символів. Функції роботи з рядками. Показчики, символні рядки, функції. Зображення рядків у програмах. Тип рядків string. Стандартні бібліотечні засоби обробки строк типу string. Вирішення прикладних задач із застосуванням рядків.

### Тема 19. Вступ до організації зовнішніх даних

Файлові структури. Фізичне та логічне подання даних на зовнішній пам'яті. Файл як тип даних. Файл у програмі та в операційній системі. Операції над файлами та над даними файлів. Файлові системи. Системні каталоги.

### Тема 20. Файлове введення/виведення

Стандартне та пряме виведення. Загальні уявлення про файлову систему. Ієрархія даних. Файли і потоки. Стандартні методи для роботи з файлами. Створення файлів послідовного доступу. Читання з файлів послідовного доступу. Приклади програм.

### Тема 21. Списки

Подання у пам'яті. Одно- та двоспрямовані списки. Організація лінійних списків. Вирішення прикладних задач із застосуванням лінійних списків.

### Тема 22. Графи

Подання у ЕОМ, операції, алгоритми, використання у програмних та інформаційних системах. Класифікація. Поняття дерева та лісу. Орієнтовані, упорядковані та бінарні дерева. Подання у ЕОМ - основні та альтернативні методи. Операції, алгоритми, використання у програмних та інформаційних системах. Орієнтовані та неорієнтовані графи. Основні визначення. Види графів та операції над графами. Представлення графів за допомогою списків суміжності та масиву ребер.

### Тема 23. Загальні положення мови С++

Основні принципи ООП. Ключові слова С++. Різниця між С та С++. Введення в класи. Конструктори та деструктори. Робота з простими класами. Показчики на об'єкти. Класи, структури, об'єднання. Різниця між класами та структурами. Масиви об'єктів класу.

## Теми практичних занять

### Практичне заняття №1-2.

Системи числення. Переклад з однієї системи числення в іншу.

### Практичне заняття №3.

Лінійні алгоритми. Алгоритми з розгалуженням. Алгоритми циклічних структур.

### Практичне заняття №4.

Базові типи даних у мовах програмування. Основні операції над даними. Арифметичні операції. Логічні та побітові операції.

### Практичне заняття №5.

Базові типи даних та введення/виведення.

### Практичне заняття №6-7.

Види умовних операторів та багатоваріантний вибір switch.

### Практичне заняття №8-9.

Оператори циклу for, while та do-while.

### Практичне заняття №10-11.

Робота з одновимірними масивами. Алгоритми роботи з послідовностями. Сортування одновимірних масивів. Алгоритми пошуку в одновимірних масивах.

### Практичне заняття №12-13.

Робота з багатовимірними масивами. Алгоритми роботи з матрицями. Робота з багатовимірними масивами.

### Практичне заняття №14-15.

Глобальні та локальні змінні. Області видимості змінних. Прототипи функцій. Розробка власних функцій. Функції користувача.

### Практичне заняття №16-17.

Рекурсія. Реалізація стандартних алгоритмів обробки масивів за допомогою рекурсії.

### Практичне заняття №18-19.

Структури. Принципи об'явлення та використання структур.

### Практичне заняття №20-21.

Показчики. Адресна арифметика.

### Практичне заняття №22.

Динамічна пам'ять. Динамічні масиви. Показчики та масиви.

### Практичне заняття №23-24.

Робота з рядками в C/C++. Функції роботи з рядками. Вирішення прикладних задач із застосуванням рядків.

### Практичне заняття №25-26.

Операції над файлами та над даними файлів. Створення файлів послідовного доступу. Читання із файлів послідовного доступу.

### Практичне заняття №27-28.

Організація лінійних списків. Вирішення прикладних задач із застосуванням лінійних списків.

### Практичне заняття №29-30.

Представлення орієнтованих та неорієнтованих графів за допомогою матриці суміжності. та інцидентності.

### Практичне заняття №31-32.

Використання класів. Конструктори та деструктори. Масиви об'єктів класу. Передача об'єктів в функції.

## Теми лабораторних робіт

### Лабораторна робота №1-2.

Робота в системі програмування Visual Studio. Налаштування і тестування програми. Створення лінійних програм. Робота з точками зупинки.

### Лабораторна робота №3-4.

Приклад простої програми з арифметичними операціями над даними. Базові типи даних та введення/виведення.

### Лабораторна робота №5.

Арифметичні операції та математичні функції мови C++. Побітові операції.

### Лабораторна робота №6-7.

Умовний оператор та оператор багатоваріантного вибору.

### Лабораторна робота №8-9.

Оператори циклу в мові C++.

### Лабораторна робота №10-11.

Алгоритми роботи з послідовностями. Робота з одновимірними масивами.

### Лабораторна робота №12-13.

Алгоритми роботи з матрицями. Робота з багатомірними масивами.

### Лабораторна робота №14-15.

Створення функцій користувача. Конструювання програм із декількох файлів.

### Лабораторна робота №16-17.

Функції. Передача масивів як параметр функції. Рекурсія.

### Лабораторна робота №18-19.

Структури. Масиви структур.

### Лабораторна робота №20-21.

Показчики та масиви. Виклик функції за посиланням. Передача параметрів за адресою. Показчики і структури.

### Лабораторна робота №22-23.

Показчики, символічні рядки, функції роботи зі строками.

### Лабораторна робота №24-25.

Форматне файлове введення-виведення.

### Лабораторна робота №26-27.

Пряме файлове введення-виведення.

### Лабораторна робота №28.

Реалізація операцій обробки лінійних односпрямованих списків.

### Лабораторна робота №29.

Реалізація операцій обробки лінійних двоспрямованих списків.

## Лабораторна робота №30.

Матричні способи представлення графів.

## Лабораторна робота №31-32.

Прості класи. Доступ до елементів класу. Конструктори та деструктори. Масиви об'єктів класу.

## Самостійна робота

Опрацювання лекційного матеріалу. Підготовка до лабораторних занять та модульних контролів. Самостійне вивчення тем та питань, які не викладаються на лекційних заняттях. Курс передбачає виконання індивідуального розрахункового завдання та курсової роботи з програмування. За результатами оформлюються письмові звіти.

## Література та навчальні матеріали

### Основна література

1. Herbert Schildt. C: The Complete Reference (Osborne Complete Reference Series). – McGraw Hill; 4th edition (May 17, 2021). ISBN-13 : 978-0072121247. – 1008 p. [Електронний ресурс]. – URL: <https://psi1.ir/wp-content/uploads/2021/09/C-The-Complete-Reference-Herbert-Schildt.pdf>
2. Balagurusamy E. Object Oriented Programming with C++. Gautam Buddha Nagar, Uttar Pradesh: Mc Graw Hill India, 2019. 537 p. [Електронний ресурс]. – URL: [https://www.anandinstitute.org/pdf/Balaguruswamy%20Object%20Oriented%20Programming%20With%20C++%20Fourth%20Edition%20\(3\).pdf](https://www.anandinstitute.org/pdf/Balaguruswamy%20Object%20Oriented%20Programming%20With%20C++%20Fourth%20Edition%20(3).pdf)
3. Трофименко О.Г., Прокоп Ю.В., Логінова Н.І., Задерейко О.В. C++. Алгоритмізація та програмування: підручник. 2-ге вид. перероб. і доповн. Одеса: Фенікс, 2019. 477 с. [Електронний ресурс]. – URL: <http://dspace.onua.edu.ua/handle/11300/11781>
4. TutorialPoint: C++ Tutorial [Електронний ресурс]. – Режим доступу: <https://www.tutorialspoint.com/cplusplus/index.htm>
5. C library [Електронний ресурс]. URL: <https://cplusplus.com/reference/clibrary/>
6. C++, C++/CLI [Електронний ресурс]. URL: [https://www.bestprog.net/uk/sitemap\\_ua/c/](https://www.bestprog.net/uk/sitemap_ua/c/)

### Додаткова література

1. Любченко Н. Ю., Соболев М.О., Паржин Ю.В., Пугачов Р.В. Основи програмування на C/C++ в прикладах. Частина 1. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2021. – 120 с. [Електронний ресурс]. – URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/54410>
2. Любченко Н. Ю., Соболев М.О., Паржин Ю.В., Пугачов Р.В., Івашко А.В. Основи програмування на C/C++ в прикладах. Частина 2. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2022. – 200 с. [Електронний ресурс]. – URL: <http://repository.kpi.kharkov.ua/handle/KhPI-Press/57740>
3. Уроки програмування на C++ : [Електронний ресурс]. URL: <https://acode.com.ua/urok-1-vvedennya-v-programuvannya/>
4. International Standard ISO/IEC 14882:2014(E) – Programming Language C++ , ISBN-13: 978-0321563842: [Електронний ресурс]. URL: <https://isocpp.org/std/the-standard>.
5. C++11 - the new ISO C++ standard: [Електронний ресурс]. URL: <https://www.stroustrup.com/C++11FAQ.html>.



## Система оцінювання

### Критерії оцінювання успішності студента та розподіл балів

Поточний контроль здійснюється у формі тестування, опитування під час лабораторних занять.

Семестровий контроль проводиться у формі іспиту у терміни, встановлені навчальним планом. Семестровий контроль проводиться в усній формі. Оцінка з іспиту може бути отримано за накопиченням балів.

Студент вважається допущеним до семестрового іспиту з навчальної дисципліни за умови виконання усіх тестових завдань, та відпрацювання обов'язкових лабораторних робіт, передбачених навчальною програмою з дисципліни.

Оцінювання проводиться за 100-бальною шкалою.

Бали нараховуються за наступним співвідношенням:

модульні тести - 20 балів; лабораторні роботи - 40 балів; розрахункове завдання (курсова робота) – 20 балів; іспит - 20 балів

### Шкала оцінювання

Сума балів	Національна оцінка	ECTS
90–100	Відмінно	A
82–89	Добре	B
75–81	Добре	C
64–74	Задовільно	D
60–63	Задовільно	E
35–59	Незадовільно (потрібне додаткове вивчення)	FX
1–34	Незадовільно (потрібне повторне вивчення)	F

### Норми академічної етики і політика курсу

Студент повинен дотримуватися «Кодексу етики академічних взаємовідносин та доброчесності НТУ «ХПІ»: виявляти дисциплінованість, вихованість, доброзичливість, чесність, відповідальність. Конфліктні ситуації повинні відкрито обговорюватися в навчальних групах з викладачем, а при неможливості вирішення конфлікту – доводитися до відома співробітників дирекції інституту. Нормативно-правове забезпечення впровадження принципів академічної доброчесності НТУ «ХПІ» розміщено на сайті: <http://blogs.kpi.kharkov.ua/v2/nv/akademichna-dobrochesnist/>

### Погодження

Силабус погоджено

27.08.2024

Завідувач кафедри  
Юрій ДОРОФЄЄВ

27.08.2024

Гарант ОП  
Марина ГРИНЧЕНКО