

## СИЛАБУС

### Кафедра автоматизованого управління технологічними процесами

<b>Назва курсу</b>	Програмування
<b>Мова викладання</b>	Українська
<b>Викладач (-і)</b>	Скрипка К.І., доцент кафедри автоматизованого управління технологічними процесами
<b>Профайл викладача</b>	Е-mail: skrypka kostyantyn@tnu.edu.ua
<b>Контакти викладача</b>	Е-mail: skrypka kostyantyn@tnu.edu.ua

#### 1. Анотація курсу

Дисципліна «Програмування» формує у студентів теоретичних знань та практичних навичок, необхідних для роботи у будь-яких адміністративних, наукових та виробничих підрозділах, в яких використовують сховища даних, а також здійснюють обслуговування та розробку інформаційних систем та сховищ даних.

Згідно з поставленою метою до задач дисципліни входить:

- Ознайомлення студентів з основними поняттями теорії засобів розробки програмного забезпечення їх призначення, загальні риси та відмінність від баз даних;
- Надання студентам знань про основні види мов програмування;
- Ознайомлення студентів з основними поняттями інтеграції даних у сховищах даних;
- Ознайомлення студентів з основними методами інтеграції даних;
- Ознайомлення студентів з основними технологіями та засобами інтеграції даних.

Короткий зміст курсу:

Здатність застосовувати знання у практичних ситуаціях.

Знання та розуміння предметної області та розуміння професії.

2.4. Передумови для вивчення дисципліни.

Передумовами вивчення дисципліни є шкільний курс «Інформатика».

- Знання з курсу «Інформатика» середньої школи.
- Основи алгоритмізації.
- Навики роботи за комп'ютером.

2.5. Результати навчання

- використовувати сучасне програмно апаратне забезпечення інформаційно комунікаційних технологій.
- використовувати програмні та програмно апаратні комплекси захисту інформаційних ресурсів.

- володіти основами конструювання ПЗ;
- володіти методами та технологіями об'єктно-орієнтованого програмування

### 3. Програма навчальної дисципліни "Основи програмування"

Змістовний модуль 1. Основні поняття програмування.

Тема 1. Основні поняття та означення

Поняття архітектури комп'ютера. Архітектури комп'ютера фон Неймана.

Архітектура системи

команд. Інформація в пам'яті комп'ютера. Типи комп'ютерів. Програмне забезпечення. Засоби

створення програм. Поняття алгоритму. Властивості алгоритмів, форми їх представлення.

Основні структури алгоритмів.

Тема 2. Синтаксис мови C++

Множина символів C++ та множина представимих символів. Правила формування констант,

ідентифікаторів. Ключові слова. Використання коментарів у програмах.

Поняття лексеми.

Змістовний модуль 2. Основні структури в програмуванні.

Тема 3. Структура програми на мові C++

Склад програми на мові C++. Вихідні файли програми. Виконання програм.

Особливості

функції main(). Поняття "час життя" і "область видимості". Приклади найпростіших програм.

Тема 4. Оголошення змінних

Базові типи даних. Правила приведення базових типів даних. Модифікатори змінних.

Автоматичні змінні. Реєстрові змінні. Зовнішні змінні та функції статичні змінні. Змінні класу

volatile. Новий стиль заголовків.

Тема 5. Керування вводом-виводом на екран

Основні оператори вводу та виводу на мові C++, зокрема на прикладі printf() та scanf(). Формат

вводу-виводу різних типів даних (чисел, рядків, вказівників тощо).

Тема 6. Вирази

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.

Змістовний модуль 3. Засоби реалізації основних структур в програмуванні.

Тема 7. Засоби мови C++ для реалізації основних структур алгоритмів

Умовні оператори. Оператори if. Оператори if-else. Умовний оператор ?..

Оператор switch.

Оператори циклу. Цикли for. Цикли while. Цикли do-while. Оператор break. Оператор continue.

Оператор goto і метки.

Тема 8. Складні типи даних в C++

Масиви. Ініціалізація масивів. Багатомірні масиви. Масиви як параметри функцій. Структури й операції з ними. Структури як аргументи функцій. Масиви структур. Показчики на структури.

Передача по посиланню членів масивів структур. Об'єднання й операції з ними.

Тема 9. Вказівники та посилання

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.

Змістовний модуль 4. Функції та робота з строковими змінними.

Тема 10. функції

Параметри й аргументи функцій. Аргументи за замовчуванням. Простір імен. Функції, що

вбудовуються (inline-). Рекурсивні функції. Математичні функції. Функції округлення.

Тема 11. Рядки і операції з ними

Масиви символів у C++. Визначення довжини рядків. Копіювання і конкатенація рядків.

Порівняння рядків. Перетворення рядків. Звертання рядків. Пошук символів.Тема 12. Використання стандартного класу стрічок string

Конструктори рядків. Зміна величини рядка і її розміру. Присвоювання, додавання і обмін

рядків. Доступ до символів рядка. Копіювання рядків і порядків. Порівняння рядків. Операції

пошуку. Вставка символів у рядок. Заміна і видалення символів з рядка. Операції вводу-

виводу рядків.

Тема 13. Файловий ввід / вивід

Взаємодія з файлами. Типи файлів. Функції стандартного вводу / виводу. Функції довільного

доступу до файлу.

## **2. Мета та цілі курсу**

Метою дисципліни «Програмування» є формування у студентів теоретичних знань та практичних навичок, необхідних для роботи у будь-яких адміністративних, наукових та виробничих підрозділах, в яких використовують сховища даних, а також здійснюють обслуговування та розробку інформаційних систем та сховищ даних.

**Значення** дисципліни для реалізації вимог кваліфікаційної характеристики фахівця та вивчення наступних дисциплін полягає в тому, що дисципліна сприяє формуванню алгоритмічного мислення майбутнього фахівця, створює базу, яка необхідна при вивченні багатьох наступних дисциплін. Виходячи з цього викладання дисципліни «Програмування» підпорядковане вирішенню таких основних задач, як з'ясування концептуальних принципів розробки ПЗ. Розглядаються питання використання комерційних компонентів для вбудованих систем.

### **3. Результати навчання**

Здатність до абстрактного мислення, аналізу та синтезу.

Здатність застосовувати знання у практичних ситуаціях.

Знання та розуміння предметної області та розуміння професійної діяльності.

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

Здатність генерувати нові ідеї (креативність).

Здатність працювати в команді.

Здатність до математичного та логічного мислення, формулювання та досліджування математичних моделей, зокрема дискретних математичних моделей, обґрунтовування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування.

Здатність до побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

Здатність опанувати сучасні методи математичного моделювання об'єктів, процесів і явищ, розробляти моделі й алгоритми чисельного розв'язування задач математичного моделювання з урахуванням похибок наближеного чисельного розв'язування професійних задач.

Здатність здійснювати формалізований опис задач дослідження операцій в організаційно-технічних і соціально-економічних системах різного призначення, визначати їх оптимальні рішення, будувати моделі оптимального вибору управління з урахуванням змін параметрів економічної ситуації, оптимізувати процеси управління в системах різного призначення та рівня ієрархії.

### **4. Обсяг курсу**

<b>Вид заняття</b>	<b>Загальна к-сть годин</b>
лекції	90
Практичні заняття	90
самостійна робота (реферат, РГР, КР, КП, тощо)	210

## 5. Пререквізити

Дисципліна «Програмування» є базовою для отримання більш глибоких знань про механізми явищ, що виникають. В значній мірі це стосується тих систем, структурна та динамічна складність яких робить неефективним чи взагалі неможливим використання аналітичних методів дослідження.

В результаті вивчення дисципліни «Програмування» студенти отримують практичні знання, але й оволодівають навичками використання пакетів прикладних програм *Microsoft Visual Studio* та ін.

## 6. Система оцінювання та вимоги

<b>Загальна система оцінювання курсу</b>	Підсумкова оцінка з дисципліни є сумою оцінок з відповідною вагою за кожен з таких видів робіт: активна робота на лабораторних та практичних заняттях, тести та підсумковий контроль (екзамен). Підсумкова оцінка визначається відповідно до поданої нижче таблиці оцінювання за різними шкалами (100-бальна, ECTS, національна).
<b>Розрахункова графічна-робота</b>	В рамках курсу не передбачено виконання РГР.
<b>Лабораторні та практичні роботи</b>	<b>Критерії оцінювання лабораторних та практичних робіт:</b> 1. Підготовленість до лабораторних/практичних занять 2. Самостійність виконання лабораторних/практичних робіт. 3. Повнота виконання завдань 4. Своєчасність виконання та захисту лабораторних/практичних робіт Максимальний бал за кожен лабораторну/практичну роботу – 5 балів
<b>Тест</b>	Проміжний тест проводиться у кожному модулі з курсу та оцінюється максимально в 10 балів.
<b>Іспит</b>	Іспит проводиться в кінці курсу, включає два теоретичних питання і одне практичне завдання. Максимально оцінюється в 40 балів.
<b>Умови допуску до підсумкового</b>	Позитивна оцінка за всіма обов'язковими видами робіт (лабораторні та практичні

## 7. Політики курсу

*Політика щодо академічної доброчесності:* списування під час тесту, іспиту заборонені.

Жодні форми порушення академічної доброчесності не толеруються. У випадку таких подій – реагування відповідно до Методичних рекомендацій для закладів вищої освіти з підтримки принципів академічної доброчесності.

*Правила перезарахування кредитів* у випадку мобільності, правила перескладання або відпрацювання пропущених занять тощо: відбувається згідно з Положення про організацію освітнього процесу у Таврійському національному університет ім. В.І. Вернадського.

*Політика щодо дедлайнів та перескладання:* роботи, які здаються із порушенням термінів без поважних причин, оцінюються на нижчу оцінку (до -50% від можливої максимальної кількості балів за вид діяльності).

*Перескладання тесту* відбувається за наявності поважних причин (наприклад, лікарняний).

*Політика щодо відвідування:* відвідування занять є обов'язковим компонентом. За об'єктивних причин (наприклад, хвороба, працевлаштування, міжнародне стажування) навчання може відбуватись в онлайн-формі за погодженням.

## РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

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. С++ Crash Course: A Fast-Paced Introduction./ Lospinoso Josh. ISBN 1593278885. - 2019.- 792с.
5. International Standard ISO/IEC 14882:2014(E) – Programming Language C++ , ISBN-13: 978-0321563842: [Електронний ресурс]. – Режим доступу: <https://isocpp.org/std/the-standard>.
6. C/C++ language and standard libraries reference: [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/en-us/library/hh875057.aspx>.

7. Мейерс С. Эффективный и современный C++. / С. Мейерс. – М.: Вильямс, 2016.