



УКРАЇНСЬКИЙ ДЕРЖАВНИЙ  
УНІВЕРСИТЕТ ЗАЛІЗНИЧНОГО  
ТРАНСПОРТУ

# АЛГОРИТМІЗАЦІЯ І ТЕХНОЛОГІЇ РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## СИЛАБУС ДИСЦИПЛІНИ

2024/2025 навчальний рік  
курс I, семестр 1, 2  
освітні програми: АКІТР, ТКРТ, ОКСКРП

Час та аудиторія проведення занять згідно розкладу:  
<http://rasp.kart.edu.ua/faculty>

**Лектор:** *Бантюков Сергій Євгенович*  
(завідувач кафедри ОТ та СУ, к.т.н., доцент)  
<https://kart.edu.ua/department/kafedra-ot-ta-su>  
Контакти: +38(057)730-10-40, +38(057)730-10-39  
e-mail: [bantyukov@kart.edu.ua](mailto:bantyukov@kart.edu.ua)  
Години консультації: *понеділок 14.00-15.00*

**Керівник практичних та лабораторних занять:**  
*Бантюков Сергій Євгенович*

Розміщення кафедри:  
м.Харків, майдан Фейєрбаха, 7, 2 корпус, 2 поверх, ауд. 2.221  
<https://kart.edu.ua/department/kafedra-ot-ta-su>  
Веб-сторінка курсу: <http://do.kart.edu.ua/>  
Додаткові інформаційні матеріали:  
<http://lib.kart.edu.ua/browse?type=author>

## Анотація курсу

У всі часи людина прагнула розширювати свої можливості, створюючи різноманітні знаряддя праці, засоби пізнання світу і полегшення існування. Так, наприклад, недостатність можливостей зору людини компенсують мікроскоп, телескоп, радіолокатор. Обмежені можливості для передачі інформації один одному розширюються телефоном, радіо, телебаченням. А обчислювальні машини доповнюють можливості людського мозку, розширюють його здатності по обробці інформації, дозволяють збільшити швидкість прийняття рішення в ході виконання будь-яких робіт.

Швидке впровадження комп'ютерів у всі сфери людської діяльності вимагає від фахівців нового покоління у галузі залізничного транспорту та інфраструктури наявності знань, умінь і практичних навичок використання новітніх комп'ютерних систем та інформаційних технологій. Сьогодні неможливо уявити ефективну роботу фахівця залізничного транспорту або управлінця залізниці без знання основ роботи комп'ютера, комп'ютерних мереж, Інтернету. Дослідження останніх років чітко показують, що комп'ютерні фундаментальні наукові дисципліни покликані визначати напрями формування і розвитку глобального інформаційного суспільства, заснованого на її знаннях.

«Алгоритмізація і технології розроблення програмного забезпечення» є комплексною науково-технічною дисципліною, що дає знання, які можуть бути застосовані для успішної роботи з технічними і програмно-апаратними засобами систем залізничної автоматики та керування різними технологічними об'єктами і процесами.

Навчальна дисципліна «Алгоритмізація і технології розроблення програмного забезпечення» розглядає як методики розробки алгоритмів обчислювальних процесів різних типів, так і основи сучасних технологій програмування, склад і можливості інтегрованих середовищ програмування. Тому головним *завданням* вивчення цієї дисципліни в нашому закладі є базова підготовка студентів в галузі використання обчислювальної техніки для вирішення інженерно-технічних та економіко-управлінських задач на залізничному транспорті.

*Метою курсу* є вивчення сучасних інформаційно-комп'ютерних технологій та застосування їх при рішенні інженерних, управлінських та організаційних задач на залізниці.

Курс має сформувати та розвинути такі *компетентності студентів*:

*Загальні компетентності:*

- здатність виявляти, ставити та вирішувати завдання за допомогою обґрунтованих рішень;
- здатність до абстрактного мислення, аналізу та синтезу, пошуку, оброблення та аналізу інформації з різних джерел у предметній галузі.

*Фахові компетентності:*

- здатність використовувати сучасні інформаційно-комунікаційні технології та програмне забезпечення для створення, аналізу, супроводження та управління проектами в ІТ-галузі;
- знання сучасних теоретичних, методичних і алгоритмічних основ розробки програмного забезпечення для його використання під час розв’язання прикладних і наукових завдань в області інформаційних систем і технологій;
- знання теоретичних та методологічних основ управління проектами різного виду в галузі комп’ютерних наук;
- здатність до розробки процедур та методів контролю ІТ-проектів;
- вміння формалізувати предметну область ІТ-проекту;
- здатність до математичного та логічного мислення;
- базова підготовка в області програмування, володіння алгоритмічним мисленням, методами програмної інженерії для реалізації програмного забезпечення з урахуванням вимог до якості та надійності інформаційних систем;
- здатність розуміти основні процеси, фази та ітерації життєвого циклу програмного забезпечення, застосувати різні види тестування та методи верифікації.

### **Чому ви маєте обрати цей курс?**

Якщо вас цікавлять інформаційні технології та комп’ютерна техніка, вам потрібно саме це!

Від здобувачів очікується знання шкільних курсів з математики, фізики, інформатики, а також базові навички володіння комп’ютером.

У результаті вивчення навчальної дисципліни Ви *будете спроможним*: розробляти алгоритми для вирішення інженерних завдань, реалізовувати їх засобами інструментального середовища (MS Visual C++); управляти роботою ПК засобами операційної системи Windows.

Використання презентацій на лекціях дає змогу надавати велику кількість наочного матеріалу, що сприяє більшому обсягу його засвоєння, а схематичне подання матеріалу – більш чіткому його структуруванню.

Під час проведення практичних та лабораторних робіт студенти мають можливість відпрацьовувати навички застосування отриманих теоретичних знань для вирішення конкретних прикладних завдань. Передбачено виконання індивідуальних завдань.

Викладачі готові надати консультації за відповідними змістовими модулями кожному студенту, в якого виникли запитання під час виконання практичних та лабораторних робіт – безпосередньо на аудиторному занятті, а під час самостійної роботи – сучасними засобами комунікацій (наприклад електронною поштою або в системі дистанційного навчання).

### **Опис навчальної дисципліни**

На дисципліну відводиться 4 кредити, модулів – 4. Загальна кількість годин – 134. На лекції відводиться 60 годин, на практичні заняття – 30 годин, на лабораторні заняття – 30 годин. Дисципліна викладається два семестри на першому курсі навчання.

### **Теми лекцій**

- Л.1. ВСТУП. Задачі курсу. КТ і науково-технічний прогрес. Історичний огляд розвитку засобів обчислювальної техніки.
- Л.2. ПОДАННЯ ІНФОРМАЦІЇ В ЕОМ. Кодування інформації. Системи числення. Перетворення чисел з однієї системи числення в іншу.
- Л.3. ПОНЯТТЯ АЛГОРИТМУ, ПРОГРАМИ. ЛІНІЙНІ ОБЧИСЛЮВАЛЬНІ ПРОЦЕСИ.
- Л.4. РОЗГАЛУЖЕНІ ОБЧИСЛЮВАЛЬНІ ПРОЦЕСИ. Логічні відношення та логічні вирази.
- Л.5. ЦИКЛІЧНІ ОБЧИСЛЮВАЛЬНІ ПРОЦЕСИ. Алгоритми обчислення суми, добутку, факторіалу. Укладенні цикли.
- Л.6. ІТЕРАЦІЙНІ ЦИКЛІЧНІ ОБЧИСЛЮВАЛЬНІ ПРОЦЕСИ. Алгоритми знаходження екстремальних значень функцій.
- Л.7. МАСИВИ. Алгоритми обробки одновимірних і двовимірних масивів.
- Л.8. АЛГОРИТМИ СОРТУВАННЯ МАСИВІВ.

Л.9. ІСТОРІЯ СТВОРЕННЯ ТА ОСОБЛИВОСТІ МОВИ C++

Л.10. ЕЛЕМЕНТАРНІ КОНСТРУКЦІЇ МОВИ C++. Ідентифікатори. Змінні. Константи. Оператори. Вирази. Основні математичні функції. Структура програми. Компілятор. Препроцесор. Директива *#include*. Директива *#define*. Оператор присвоєння.

Л.11. СТАНДАРТНЕ ВВЕДЕННЯ-ВИВЕДЕННЯ ДАНИХ. Функція стандартного введення *scanf()*. Функція стандартного виведення *printf()*.

Л.12. УПРАВЛІННЯ ОБЧИСЛЮВАЛЬНИМ ПРОЦЕСОМ. Оператор умовної передачі управління *if*. Укладені оператори *if*. Оператор безумовної передачі управління *goto*. Оператор *switch*.

Л.13. ЦИКЛІЧНІ ОБЧИСЛЮВАЛЬНІ ПРОЦЕСИ. Цикл *for*. Цикл *while*. Цикл *do-while*. Оператор *break*. Оператор *continue*.

Л.14. ПРОГРАМУВАННЯ ОБРОБКИ ОДНОВИМІРНИХ МАСИВІВ. Рядки. Функція *gets()*. Функція *puts()*. Ініціалізація безрозмірних символічних масивів.

Л.15. ПРОГРАМУВАННЯ ОБРОБКИ ДВОВИМІРНИХ МАСИВІВ. Ініціалізація двовимірних масивів. Ініціалізація безрозмірних числових масивів.

Л.16. СТРУКТУРИ. Поняття структури. Звернення до елементів структури. Оператор *.* (крапка). Масиви структур.

Л.17. ФУНКЦІЇ. Основні поняття. Визначення функції. Прототип функції. Виклик функції.

Л.18. КЛАСИ ПАМ'ЯТІ. Час життя та область видимості програмних об'єктів. Логічна структура пам'яті програми. Динамічні, автоматичні, регістрові, статичні, зовнішні змінні. Оголошення змінних. Оголошення функцій. Ініціалізація глобальних та локальних змінних.

Л.19. ПОКАЖЧИКИ. Показчик-змінна. Оператори з покажчиками. Вирази з покажчиками. Показчики і масиви. Показчики і рядки. Масив покажчиків. Ініціалізація покажчиків. Функції обробки рядків з використанням покажчиків.

Л.20. ВВЕДЕННЯ-ВИВЕДЕННЯ ЗА ДОПОМОГОЮ ПОТОКІВ STL.

Л.21. ПРОСТІР ІМЕН В C++.

Л.22. Перевантаження функцій у C++.

Л.23. ОБ'ЄКТНО-ОРІЄНТОВАНИЙ ПІДХІД. Основні поняття об'єктно-орієнтованого підходу.

Л.24. Класи у мові C++.

Л.25. Конструктори та деструктори.

Л.26. Інкапсуляція. Поліморфізм. Успадкування.

## Теми лабораторних та практичних робіт

1. Проектування алгоритмів лінійних обчислювальних процесів.
2. Проектування алгоритмів розгалужених обчислювальних процесів.
3. Проектування алгоритмів циклічних обчислювальних процесів: знаходження сум, добутків, факторіалів.
4. Проектування алгоритмів ітераційних циклічних обчислювальних процесів.
5. Проектування алгоритмів обробки одновимірних масивів.
6. Проектування алгоритмів обробки двовимірних масивів.
7. Проектування алгоритмів сортування масивів.
8. Програмування алгоритмів лінійних обчислювальних процесів.
9. Програмування алгоритмів розгалужених обчислювальних процесів.
10. Програмування алгоритмів циклічних обчислювальних процесів: знаходження сум, добутків, факторіалів.
11. Програмування алгоритмів ітераційних циклічних обчислювальних процесів.
12. Програмування алгоритмів обробки одновимірних масивів.
13. Програмування алгоритмів обробки двовимірних масивів.
14. Програмування алгоритмів сортування масивів.
15. Програмування структур, масивів структур.
16. Програмування задач з використанням покажчиків.
17. Програмування задач з використанням об'єктно-орієнтованого підходу.

### Ресурси курсу

Електронна бібліотека УкрДУЗТ. [Електронний ресурс]. – Режим доступу: <http://metod.kart.edu.ua/dsearch/process/page/1/fid/2/sf/0/aid/18/authors//title//key/>

Матеріали дистанційного навчання. УкрДУЗТ. [Електронний ресурс]. – Режим доступу: <http://do.kart.edu.ua/>

InfoCiti. [Електронний ресурс]. – Режим доступу: <http://www.infocity.kiev.ua>

Національна бібліотека України імені В.І. Вернадського. [Електронний ресурс]. – Режим доступу: <http://www.nbuv.gov.ua>

Мислене древо (Українські інформаційні ресурси для освіти і науки). [Електронний ресурс]. – Режим доступу: <https://www.myslenedrevo.com.ua/>

Visual Studio IDE, Code Editor, Azure DevOps, & App Center <https://visualstudio.microsoft.com > subscriptions>

## Контрольні заходи результатів навчання

Вивчення навчальної дисципліни «Алгоритмізація і технології розроблення програмного забезпечення» потребує:

- виконання завдань згідно з навчальним планом (індивідуальні, домашні завдання, самостійна робота тощо);
- підготовки до практичних та лабораторних занять;
- роботи з інформаційними джерелами.

Поточний контроль, модульний контроль (тести), підсумкове тестування, залік, іспит. При оцінюванні результатів навчання викладач керується Положенням про контроль та оцінювання якості знань студентів в УкрДУЗТ (<http://kart.edu.ua/images/stories/akademiya/documentu-vnz/polojennya12-2015.pdf>).

Згідно Положенню про впровадження кредитно-модульної системи організації навчального процесу використовується 100-бальна шкала оцінювання.

Принцип формування оцінки за перший та другий залікові модулі відбувається за 100-бальною шкалою, що наведено у таблиці, де максимальна кількість балів, яку може набрати студент за різними видами навчального навантаження.

Вид роботи	Модуль	
	I	II
<b>Виконання та захист практичних та лабораторних робіт, робота на лекціях, самостійне виконання індивідуальних завдань</b>	60	60
<b>Модульне тестування</b>	40	40
<b>Сума балів за модуль</b>	100	100
<b>Підсумковий контроль</b>	<b>100 (іспит)</b>	

Модульне тестування – комп'ютерне тестування наприкінці модуля, яке містить завдання по всьому вивченому за модуль матеріалу. Оцінюються за вірними відповідями на тестові модульні питання (не менш 10 запитань в тесті, кожна вірна відповідь оцінюється до 4 балів). **Максимальна кількість становить 40 балів за модуль.**

Відвідування лекцій: На лекціях та лабораторних заняттях присутність здобувачів вищої освіти є обов'язковою, важливою також є їх участь в обговоренні всіх питань з теми. Пропущені заняття мають бути відпрацьовані. Це ж стосується й студентів, які не виконали завдання або показали відсутність знань з основних питань теми. Здобувач вищої освіти повинен дотримуватися навчальної етики, поважно ставитися до учасників процесу навчання, бути зваженим, уважним та дотримуватися дисципліни і плану навчального процесу. Бали за цю складову не нараховуються взагалі, якщо студент не відвідував більш 50% лекційних занять у модулі без поважних причин. За відвідування кожної лекції нараховується 1 бал. **Максимальна сума становить 15 балів.**

Ступінь залученості: Мета участі в курсі – залучити студента до дискусії, щоб він мав розширити можливості навчання для себе та своїх однолітків, та дати йому ще один спосіб перевірити свої погляди на питання застосування принципів алгоритмізації та технологій програмування у майбутній фаховій діяльності. Участь буде оцінюватися на основі кількості та вірності його відповідей. **Максимальна сума становить 10 балів.**

Практичні заняття: Підготовка до практичних занять передбачає: ознайомлення із програмою навчальної дисципліни, питаннями, які виносяться на заняття з відповідної теми; вивчення навчального матеріалу. Оцінюються за відвідуваннями (до 3 балів), ступенем залученості (до 7 балів) та якістю виконання роботи (до 5 балів). **Максимальна сума становить 15 балів.**

Лабораторні заняття: Підготовка до лабораторних занять передбачає: ознайомлення із програмою навчальної дисципліни, питаннями, які виносяться на заняття з відповідної теми; вивчення навчального матеріалу. Оцінюються за відвідуваннями (до 3 балів), ступенем залученості (до 7 балів) та якістю виконання роботи (до 5 балів). **Максимальна сума становить 15 балів.**

Завдання на самостійну роботу: Вирішення індивідуальних, домашніх завдань повинно відповідати (мати всі необхідні складові), як за формою, так і за змістом, вимогам, що висуваються до вирішення відповідного завдання, свідчити про його самостійність (демонструвати ознаки самостійності виконання здобувачем такої роботи), відсутність ознак повторюваності та плагіату.

Залік, іспит. Студент отримує оцінку із заліку або іспиту за результатами першого та другого модульного контролю шляхом накопичення балів. Максимальна кількість балів, яку може отримати студент становить 100 (до 60 балів поточного контролю та до 40 балів тестування). Середнє арифметичне суми модульних оцінок складає бал заліку або іспиту. За бажанням студента остаточний бал іспиту може бути підвищений з В до А або з D до C під час здачі заліку або іспиту.

При заповненні заліково-екзаменаційної відомості та залікової книжки (індивідуального навчального плану) студента, оцінка виставляється за шкалою ECTS із вказаною кількістю набраних балів та відповідною буквою.



<b>За шкалою ECTS</b>	<b>За 100 бальною шкалою</b>	<b>ECTS оцінка</b>
<b>Відмінно – відмінне виконання лише з незначною кількістю помилок</b>	90-100	A
<b>Дуже добре – вище середнього рівня з кількома помилками</b>	82-89	B
<b>Добре – в загальному правильна робота з певною кількістю грубих помилок</b>	75-81	C
<b>Задовільно - непогано, але зі значною кількістю недоліків</b>	69-74	D
<b>Достатньо – виконання задовольняє мінімальні критерії</b>	60-68	E
<b>Незадовільно – потрібно попрацювати перед тим як отримати залік або іспит (без повторного вивчення модуля)</b>	35-59	FX
<b>Незадовільно - необхідна серйозна подальша робота (повторне вивчення модуля)</b>	<35	F

До перелічених складових модульної оцінки можуть нараховуватися додаткові бали за участь студента у науковій роботі, підготовці публікацій, робіт на конкурси, участь в олімпіадах тощо. Кількість додаткових балів визначається на розсуд викладача, але у сумі не більш 100 балів разом з переліченими складовими модульної оцінки. Обґрунтованість нарахування студенту додаткових балів розглядається на засіданні кафедри та оформлюється відповідним протоколом.

### **Кодекс академічної доброчесності**

Порушення Кодексу академічної доброчесності Українського державного університету залізничного транспорту є серйозним порушенням, навіть якщо воно є ненавмисним. Кодекс доступний за посиланням: <http://kart.edu.ua/documentu-zvo-ua>

Зокрема, дотримання Кодексу академічної доброчесності УкрДУЗТ означає, що вся робота на іспитах та заліках має виконуватися індивідуально. Під час виконання самостійної роботи студенти можуть консультуватися з викладачами та з іншими студентами, але повинні самостійно розв'язувати завдання, керуючись власними знаннями, уміннями та навичками. Посилання на всі ресурси та джерела (наприклад, у звітах, самостійних роботах чи презентаціях) повинні бути чітко визначені та оформлені належним чином. У разі спільної роботи з іншими студентами над виконанням індивідуальних завдань, викладач оцінює ступінь залученості до роботи кожного студента.

## **Інтеграція студентів із обмеженими можливостями**

Вища освіта є провідним чинником підвищення соціального статусу, досягнення духовної, матеріальної незалежності і соціалізації молоді з обмеженими функціональними можливостями й відображає стан розвитку демократичних процесів і гуманізації суспільства. Для інтеграції студентів із обмеженими можливостями в освітній процес Українського державного університету залізничного транспорту створена система дистанційного навчання на основі сучасних педагогічних, інформаційних, телекомунікаційних технологій. Доступ до матеріалів дистанційного навчання з цього курсу можна знайти за посиланням: <http://do.kart.edu.ua/>

## **Очікувані результати навчання**

В результаті навчання студент *відпрацьовує навик* виявляти, ставити та вирішувати проблеми за допомогою обґрунтованих рішень, *має розвинути здатність* використовувати сучасні інформаційно-комунікаційні технології та програмне забезпечення для створення, аналізу, супроводження та управління проектами в ІТ-галузі; абстрактного мислення, аналізу та синтезу, пошуку, оброблення та аналізу інформації з різних джерел у будівництві; *набути знання* сучасних теоретичних, методичних і алгоритмічних основ розробки програмного забезпечення для його використання під час розв'язання прикладних і наукових завдань в області інформаційних систем і технологій; теоретичних та методологічних основ управління проектами в галузі комп'ютерних наук; формалізації предметної області, розробки процедур та методів контролю та управління якістю та вимогам ІТ-проектів; здатність до математичного та логічного мислення; ґрунтовна підготовка в області програмування, володіння алгоритмічним мисленням, методами програмної інженерії для реалізації програмного забезпечення з урахуванням вимог до якості та надійності інформаційних систем; здатність розуміти основні процеси, фази та ітерації життєвого циклу програмного забезпечення, застосувати різні види тестування та методи верифікації.