

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра комп'ютерних систем, мереж і кібербезпеки (№ 503)

ЗАТВЕРДЖУЮ

Гарант освітньої програми


(підпис)

Анатолій ШОСТАК
(ім'я та ПРІЗВИЩЕ)

« 31 » серпня 2025 р.

**СИЛАБУС ОBOB'ЯЗKОВОЇ
HABЧАЛЬНОЇ ДИСЦИПЛІНИ**

Технології програмування
(назва навчальної дисципліни)

Галузь знань: F «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: F7 «Комп'ютерна інженерія»
(код і найменування спеціальності)

Освітня програма: «Системне програмування»
(найменування освітньої програми)

Рівень вищої освіти: перший (бакалаврський)

Силабус введено в дію з 01.09.2025

Харків – 2025 р.

Розробник (и): Бабешко Є.В., доцент, к.т.н., доцент
(прізвище та ініціали, посада, науковий ступінь і вчене звання)


(підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри _____
комп'ютерних систем, мереж і кібербезпеки
(назва кафедри)

Протокол № 1 від « 29 » серпня 2025 р.

Завідувач кафедри д.т.н., професор
(науковий ступінь і вчене звання)


(підпис)

Вячеслав ХАРЧЕНКО
(ім'я та ПРІЗВИЩЕ)

Погоджено з представником здобувачів освіти:


(підпис)

Дар'я ФЕДОРЕНКО
(ім'я та ПРІЗВИЩЕ)

1. Загальна інформація про викладача



ПІБ: Бабешко Євген Васильович

Посада: Доцент

Науковий ступінь: Кандидат технічних наук

Вчене звання: Доцент

Перелік дисциплін, які викладає:

Індустріальні IoT системи
Безпека індустріальних систем та Інтернету речей
Технології програмування

Напрями наукових досліджень:

Моделювання та оцінювання надійності та безпеки, розроблення та верифікація програмного забезпечення для критичних галузей, індустріальний Інтернет речей.

Контактна інформація:

e.babeshko@csn.khai.edu

2. Опис навчальної дисципліни

Форма здобуття освіти	Денна
Семестр	2
Мова викладання	Українська
Тип дисципліни	Обов'язкова
Обсяг дисципліни: кредити ЄКТС/ кількість годин	4,5 кредити ЄКТС / 135 годин (64 аудиторних, з яких: лекції – 32, лабораторні – 32; СРЗ – 71)
Види навчальної діяльності	Лекції, лабораторні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	Дисципліна базується на знаннях, отриманих під час вивчення дисциплін у циклі загальної і професійної підготовки, передбачених навчальним планом спеціальності. Матеріал дисципліни базується на знаннях, отриманих під час вивчення дисциплін «Технології програмування» (І семестр), «Українська мова за професійним спрямуванням».

3. Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – надання необхідних знань з структурного програмування, а також формування твердих практичних навичок щодо розроблення програмного забезпечення з використанням структурного підходу, а також засвоєння основних положень структурного принципу при створенні комп'ютерних програм, вивчення мови програмування високого рівня.

Завдання – придбання здобувачами необхідних знань та вмінь в сфері проєктування програмного забезпечення на основі відповідного набору абстрактних типів даних, вирішення прикладних задач з використанням стандартних типів даних і типів даних власного розроблення, отримання навиків використання ключових концепцій структурного підходу, а також вивчення засобів і основних принципів побудови алгоритмів, створення та використання структур даних, вивчення синтаксису мови програмування C.

Компетентності, які набуваються:

Загальні компетентності (ЗК):

- ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.
- ЗК3. Здатність застосовувати знання у практичних ситуаціях.
- ЗК7. Вміння виявляти, ставити та вирішувати проблеми.

Фахові компетентності (ФК):

- ФК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.
- ФК5. Здатність використовувати засоби і системи автоматизації проєктування до розроблення компонентів комп'ютерних систем та мереж, Інтернет додатків, кіберфізичних систем тощо.
- ФК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

Програмні результати навчання (ПРН):

- ПРН1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.
- ПРН3. Знати новітні технології в галузі комп'ютерної інженерії.
- ПРН4. Знати та розуміти вплив технічних рішень в суспільному, економічному, соціальному і екологічному контексті.
- ПРН6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.
- ПРН7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності.

- ПРН8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей.
- ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.
- ПРН13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів.
- ПРН16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.
- ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.
- ПРН20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

4. Зміст навчальної дисципліни

Семестр 2

МОДУЛЬ 1

Змістовний модуль 1. Модульність, тестування та організація програм

Тема 1. Вступ

Стисла анотація: Інформація про мету та обсяг дисципліни. Принципи оцінювання. Рекомендована література та веб-ресурси. Умовний оператор. Стандартні типи даних. Створення користувацьких типів даних. Оператор sizeof (повторення).

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 1

Тема 2. Модулі

Стисла анотація: Вихідні файли. Переваги розбиття програм. Заголовні файли. Директива #include. Спільний доступ до прототипів функцій та змінних. Відкриті та закриті функції. Приклад реалізації модуля.

Лабораторна робота №1: Розгалуження та вибір

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 2. Заголовні файли для реалізації драйверів.

Тема 3. Модульне тестування

Стисла анотація: Класифікація тестування за масштабом. Тестувальні модулі. Використання Google Test Framework. Контрактне програмування. Припущення. Заголовний файл assert.h. Заголовний файл errno.h. Функції perror та strerror. Розроблення через тестування.

Лабораторна робота №2: Модульне тестування

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 3. Робота з Google Test Framework

Тема 4. Масиви та функції: приклади практичного використання

Стисла анотація: Ініціалізація масивів. Використання sizeof з масивами. Константні масиви. Масиви як аргументи функцій.

Лабораторна робота №3: Використання масивів для завдань системного програмування

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 4. Використання масивів для представлення статусу пристроїв.

Тема 5. Відлагодження

Стисла анотація: Різновиди помилок. Точки зупинки. Призначення і способи використання відлагоджувача середовища Microsoft Visual Studio. Перегляд даних у відлагоджувачі. Покрокове виконання. Виконання до заданого рядка. Продовження виконання.

Лабораторна робота №4: Функції. Робота з відлагоджувачем

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 5. Використання відлагоджувача для відлагодження драйверів.

Тема 6. Рекурсія

Стисла анотація: Рекурсивні функції. Швидке сортування.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 6. Використання рекурсивних функцій для задач системного програмування.

Тема 7. Організація програми

Стисла анотація: Локальні змінні. Статичні локальні змінні. Параметри функції. Глобальні змінні. Блоки. Захист від некоректного введення.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 7. Типова структура та складові драйверу.

МОДУЛЬ 2

Змістовний модуль 2. Пам'ять, структури даних, файлове введення/виведення

Тема 8. Вказівники

Стисла анотація: Змінні-вказівники. Оператор взяття адреси. Оператор розіменування. Присвоювання вказівників. Вказівники як аргументи функцій. Повернення вказівників з функцій.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 8. Використання вказівників у функціях ядра.

Тема 9. Вказівники та масиви

Стисла анотація: Арифметика вказівників. Вказівники для оброблення масивів. Ім'я масиву як вказівник. Масиви як аргументи. Вказівник як ім'я масиву. Вказівники та багатовимірні масиви.

Лабораторна робота №5: Застосування вказівників

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 9. Використання вказівників на константні масиви.

Тема 10. Рядки

Стисла анотація: Рядкові літерали. Рядкові змінні. Введення та виведення рядків. Функції заголовного файлу string.h. Аргументи командного рядка.

Лабораторна робота №6: Рядки. Параметри командного рядку

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 10. Робота з Simple Dynamic Strings.

Тема 11. Структури, об'єднання, переліки

Стисла анотація: Структури. Структурні змінні. Структурні типи даних. Вкладені структури. Об'єднання. Переліки.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 11. Структурні типи даних для завдань системного програмування.

Тема 12. Динамічне виділення пам'яті

Стисла анотація: Нульові вказівники. Динамічні рядки. Динамічні масиви. Звільнення пам'яті. Зв'язні списки.

Лабораторна робота №7: Структури, текстові та двійкові файли, динамічне виділення пам'яті

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 12. Динамічні структури даних.

Тема 13. Робота з файлами

Стисла анотація: Потоки. Текстові та двійкові файли. Операції над файлами. Виведення у файли. Введення з файлів. Функції для роботи з файлами. Оброблення помилок при роботі з файлами.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 13. Створення лог-файлів.

Тема 14. Абстрактні об'єкти та абстрактні типи даних

Стисла анотація: Різновиди модулів. Реалізація стеку з використанням масиву. Реалізація стеку з використанням зв'язного списку. Реалізація стеку з абстрактного типу даних.

Самостійна робота: Відпрацювання матеріалів лекційних занять за темою 14. Створення абстрактних типів даних «канал» та «сигнал»

Розрахункова робота

5. Індивідуальні завдання

Тематика розрахункових робіт:

1. Збереження стану програми при завершенні роботи та його відновлення при повторному запуску
2. Використання динамічних структур даних при роботі з файлами
3. Використання модулів для створення абстрактних об'єктів
4. Використання модулів для створення абстрактних типів даних

6. Методи навчання

Проведення аудиторних лекцій, лабораторних робіт, консультацій, а також самостійна робота здобувачів з використанням відповідних матеріалів (п.11, 12).

7. Методи контролю

Проведення поточного контролю, електронного тестування, підсумковий контроль у вигляді іспиту.

8. Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Виконання і захист лабораторних робіт	0...10	4	0...40
Проходження тестів для перевірки знань	0...5	1	0...5
Модульний контроль	0...10	1	0...10
Змістовний модуль 2			
Виконання і захист лабораторних робіт	0...10	3	0...30
Проходження тестів для перевірки знань	0...5	1	0...5
Розрахункова робота	0...5	1	0...5
Модульний контроль	0...5	1	0...5
Усього за семестр			0...100

Семестровий контроль (іспит) проводиться у разі відмови здобувача освіти від балів підсумкового контролю й за наявності допуску до іспиту. Під час складання семестрового іспиту здобувач освіти має можливість отримати максимум 100 балів.

Білет для іспиту складається з двох теоретичних запитань (максимальна кількість балів за кожне – 25), тесту (максимальна кількість балів – 25) та практичного запитання (максимальна кількість балів – 25).

Таблиця 8.2 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційний залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

Критерії оцінювання роботи здобувача освіти протягом семестру

Задовільно (60-74) – Показати мінімум знань та умінь. Захистити не менше 75% від усіх завдань лабораторних занять. Знати можливості та основні положення роботи з мовою програмування C. Знати базові структури даних. Знати основи роботи з середовищем Microsoft Visual Studio. Уміти створювати проекти у середовищі Microsoft Visual Studio. Уміти створювати консольні програми з використанням мови програмування C.

Добре (75-89) – Твердо знати мінімум, захистити не менше 90% завдань лабораторних занять. Знати ключові принципи структурного програмування. Уміти розробляти алгоритми та документувати їх у вигляді схем алгоритмів. Уміти реалізовувати програмне оброблення файлів.

Відмінно (90-100) – Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти їх застосовувати. Уміти розбивати програми на модулі. Уміти використовувати відлагоджувач. Уміти створювати нові типи даних. Уміти розробляти модульні тести.

9. Політика навчального курсу

Відвідування занять. Інтерактивний характер курсу передбачає обов’язкове відвідування лабораторних занять. Здобувачі освіти, які за певних обставин не можуть відвідувати лабораторні заняття регулярно, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після їх пропуску. Відпрацювання занять здійснюється усно у формі співбесіди за питаннями, визначеними планом заняття. В окремих випадках дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни

здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>). Очікується, що роботи здобувачів освіти будуть їх оригінальними дослідженнями. Відсутність посилань на використані джерела, фабрикавання джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu/ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

10. Методичне забезпечення

1. Сторінка дисципліни у системі дистанційного навчання «Ментор» [Ел. ресурс]. URL: <https://mentor.khai.edu/course/view.php?id=5162>

11. Рекомендована література

Базова

1. Ришковець Ю.В., Висоцька В.А. Алгоритмізація та програмування. Частина 1: навч. посібник. Львів: Новий світ - 2000, 2021. 337 с.
2. Іванов Є.О., Ліндер Я.М., Жереб К.А. Основи мови програмування C++: навчальний посібник. К.: Логос, 2020. 90 с.
3. Галісеєв Г. Системне програмування. К: Університет «Україна», 2019. 113 с.
4. Шпак З.Я. Програмування мовою C: навч. посіб. / З.Я. Шпак; Нац. ун-т «Львів. політехніка». 2-ге вид., допов. Львів: Видавництво Львівської політехніки, 2011. 436 с.

Допоміжна

1. Козак Л. І., Костюк І. В., Стасевич С. П. Основи програмування: навчальний посібник – Львів: «Новий Світ-2000», 2020. 328 с.
2. King K.N. C Programming. A Modern Approach / K.N. King. 2nd edition. W.W. Norton & Company, 2008. 832 p.

12. Інформаційні ресурси

1. Підручник C. [Ел. ресурс]. URL: <https://w3schoolsua.github.io/c/index.html>
2. GoogleTest – Google Testing and Mocking Framework. [Ел. ресурс]. URL: <https://github.com/google/googletest>
3. Simple Dynamic Strings. [Ел. ресурс]. URL: <https://github.com/antirez/sds>
4. Modern C [Ел. ресурс]. URL: <https://inria.hal.science/hal-02383654>
5. Stackoverflow. Питання з міткою «C» [Ел. ресурс]. URL: <https://stackoverflow.com/questions/tagged/c>
6. Beej's Guide to C Programming [Ел. ресурс]. URL: <https://beej.us/guide/bgc/>
7. Getting started with drivers on Windows [Ел. ресурс]. URL: <https://learn.microsoft.com/en-us/windows-hardware/drivers/gettingstarted/>