



# АЛГОРИТМИ ТА СТРУКТУРИ ДАНИХ-1.

## ОСНОВИ АЛГОРИТМІЗАЦІЇ

### Робоча програма навчальної дисципліни (Силабус)

#### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>121 Інженерія програмного забезпечення</i>
Освітня програма	<i>Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем і веб-технологій</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Заочна</i>
Рік підготовки, семестр	<i>1 курс, осінній семестр</i>
Обсяг дисципліни	<i>105 годин / 3,5 кредита ЕКТС (лекції – 4 год., лабораторні заняття – 4 год., СРС – 97 год.)</i>
Семестровий контроль/ контрольні заходи	<i>Залік, МКР</i>
Розклад занять	<i><a href="http://rozklad.kpi.ua/">http://rozklad.kpi.ua/</a></i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лекції і лабораторні заняття проводить: канд. техн .наук, доцент, Кублій Лариса Іванівна <a href="mailto:kublji_l_i@ukr.net">kublji_l_i@ukr.net</a>, тел. 544-50-15, 063-71-91-231, 097-558-27-17</i>
Розміщення курсу	<i>Кампус</i>

#### Програма навчальної дисципліни

##### 1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Застосування існуючих і розробка нових алгоритмів є невід’ємною складовою роботи кожного програміста. Алгоритми є основою програмування і вони використовуються в усіх сферах, пов’язаних з застосуванням інформаційних технологій. Для розв’язання нових задач і розробки нових алгоритмів потрібно знати основні алгоритми розв’язування типових задач. У курсі “Алгоритми та структури даних-1. Основи алгоритмізації”, який студенти спеціальності 121 “Інженерія програмного забезпечення” вивчають на першому курсі в першому семестрі, розглянуто як класичні, так і сучасні алгоритми, зокрема, алгоритм Евкліда знаходження найбільшого спільного дільника, решето Ератосфена пошуку простих чисел, схема Ньютона для обчислення коренів, алгоритми генерації випадкових чисел, генерації комбінаторних множин, алгоритми, які базуються на використанні кодів Грея і використанні побітових операцій, досить велику увагу приділено точності обчислень при роботі з дійсними числами. Алгоритми, які вивчаються в даному курсі, в основному працюють з простими структурами даних. Розглянуто питання дослідження ефективності алгоритмів.

**Метою** опанування дисципліни є формування у студентів професійних компетентностей:

- здатність до абстрактного мислення, аналізу та синтезу (ЗК 1);
- здатність застосовувати знання у практичних ситуаціях (ЗК 2);

- здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК 6);
- володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних. (ФК 7);
- здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення (ФК 8);
- здатність до алгоритмічного та логічного мислення (ФК 14);
- здатність розробляти ефективні алгоритми для застосування при розв'язанні задач залежно від предметного середовища, застосовувати алгоритми для конкретних задач, перетворювати алгоритми в програмний код;
- здатність використовувати математичні методи для прийняття ефективних рішень під час розв'язання професійних задач в процесі розробки ІС та ІТ;
- здатність використання принципів структурного програмування, основних структур даних під час реалізації алгоритмів професійних завдань.

**Предметом** навчальної дисципліни є вивчення простих структур даних і типових алгоритмів роботи з ними.

Студенти після засвоєння кредитного модуля мають продемонструвати такі результати навчання:

#### **знання**

- сучасних методів побудови та аналізу ефективних алгоритмів, оцінку складності алгоритмів;
- теоретичних основ алгоритмізації і проектування програм;
- методів структурного програмування;

#### **уміння**

- аналізувати, цілеспрямовано шукати і вибирати необхідні для розв'язання професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки (ПРН 1);
- вибирати вихідні дані для проектування, керуючись формальними методами опису вимог та моделювання (ПРН 11);
- знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань (ПРН 13);
- розробляти, аналізувати та застосовувати ефективні алгоритми для розв'язання професійних завдань в галузі інформаційних технологій;
- володіти методами та технологіями організації та застосування даних програмного забезпечення, а саме:
  - застосовувати мови програмування;
  - використовувати рекурсивні алгоритми;
  - аналізувати та програмувати виключні ситуації у програмних продуктах;
  - розробляти та використовувати алгоритми сортування та пошуку;
  - розробляти прикладне програмне забезпечення комп'ютерних інформаційних систем;
  - застосовувати основні структури даних під час програмної реалізації алгоритмів;
  - застосовувати ефективні алгоритми для розв'язання професійних завдань;
  - проектувати елементи математичного та лінгвістичного забезпечення обчислювальних систем;
  - застосовувати типові алгоритми обробки даних.

#### **Додаткові знання:**

- особливостей предметного середовища, застосування відповідних структур даних, спеціалізованих алгоритмів, різних мовних конструкцій, парадигм та технік програмування та розроблення програмного забезпечення, з особливостей організації та використання різних структур даних.

#### **Додаткові вміння:**

- конструювати, кодувати, тестувати, налагоджувати та супроводжувати програмний код.

## **2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)**

У структурно-логічній схемі навчання зазначена дисципліна розміщена у першому семестрі першого року навчання. Дисципліна “Алгоритми та структури даних-1. Основи алгоритмізації” не має дисциплін, які її забезпечують, вона вивчається на основі знань, отриманих у середній загальноосвітній школі, а також паралельно з такими дисциплінами, як “Основи програмування”, “Комп’ютерна дискретна математика”, “Вища математика”. Компетенції, отримані студентами в процесі вивчення модуля застосовуються при вивченні таких навчальних дисциплін як: “Бази даних”, “Компоненти програмної інженерії”, “Об’єктно-орієнтований аналіз та конструювання програмних систем”, “Архітектура системного програмного забезпечення”, “Безпека програмного забезпечення”, “Основи розробки трансляторів”, “Основи інженерії даних”.

## **3. Зміст навчальної дисципліни**

**Тема 1.** Поняття алгоритму. Подання алгоритмів. Типові алгоритмічні структури програмування

**Тема 2.** Позиційні і непозиційні системи числення

**Тема 3.** Лінійні алгоритми

**Тема 4.** Алгоритми з розгалуженнями

**Тема 5.** Циклічні алгоритми

**Тема 6.** Модульна структура програми. Рекурсивні алгоритми

**Тема 7.** Алгоритми роботи з масивами

**Тема 8.** Алгоритмізація типових обчислювальних задач

## **4. Навчальні матеріали та ресурси**

### **Базова література**

1. Ахо А., Хопкрофт Д., Ульман Д. Структуры данных и алгоритмы: Пер. с англ. — М.: Изд. дом “Вильямс”, 2019. — 400 с.
2. Вирт Н. Алгоритмы + структуры данных = программы. — М.: Мир, 1985. — 406 с.
3. Кнут Д. Искусство программирования. Основные алгоритмы. Том 1. — М.: Изд. дом “Вильямс”, 2002. — 720 с.
4. Кнут Д. Искусство программирования. Получисленные алгоритмы. Том 2. — М.: Изд. дом “Вильямс”, 2001. — 832 с.
5. Кнут Д. Искусство программирования. Комбинаторные алгоритмы. Том 4А. — М.: Изд. дом “Вильямс”, 2013. — 960 с.
6. Кублій Л. І. Алгоритмізація та програмування. Практикум [Електронний ресурс] : навчальний посібник для здобувачів ступеня бакалавра за спеціальністю 122 “Комп’ютерні науки” / Л. І. Кублій; КПІ ім. Ігоря Сікорського. — Електронні текстові дані (1 файл: 28,15 Мбайт). — Київ: КПІ ім. Ігоря Сікорського, 2019. — 209 с. — [ela.kpi.ua/handle/123456789/28216](http://ela.kpi.ua/handle/123456789/28216)
7. Новотарський М.А. Алгоритми та методи обчислень [Електронний ресурс] : навч. посіб. для студ. спеціальностей 121 “Інженерія програмного забезпечення”, спеціалізації “Програмне забезпечення високопродуктивних комп’ютерних систем та мереж” та 123 “Комп’ютерна інженерія”, спеціалізації “Комп’ютерні системи та мережі”. — Електронні текстові дані (1 файл: 4648 Кбайт). — Київ : КПІ ім. Ігоря Сікорського, 2019. — 407 с. — [https://ela.kpi.ua/bitstream/123456789/27864/1/Alhorytmy\\_ta\\_metody\\_obchislenn.pdf](https://ela.kpi.ua/bitstream/123456789/27864/1/Alhorytmy_ta_metody_obchislenn.pdf) .
8. Хайнеман Дж., Пояяис Г., Сеяков С. Алгоритмы. Справочник с примерами на C, C++, Java и Python, 2-е изд.: Пер. с англ. — СПб.: ООО “Альфа-книга”, 2017. — 432 с.

### **Додаткова література**

9. Агеев М.И., Алик В.П., Галис Р.М., Марков Ю.И. Библиотека алгоритмов 16-50б (Справочное пособие). — М.: Советское радио, 1975. — 176 с. (С. 43).
10. Бородкіна І.Л., Бородкін Г.О. Теорія алгоритмів: Посібник для студентів вищих навчальних закладів. — К.: Національний університет біоресурсів та природокористування України, 2018. — 231 с.
11. Дасгупта С., Пападимитриу Х., Вазирані У. Алгоритмы. — М. : МЦНМО, 2014. — 320 с.
12. Криптография от папируса до компьютера. — <http://www.taina.dinamikweb.ru/a/index.htm> (<http://www.taina.dinamikweb.ru/a/index-prostejshie-algoritmy-generacii-chast-2.htm>)
13. Львов М.С., Співаковський О.В. Основи алгоритмізації та програмування: Навч. посібник. — Херсон, 1997. — 371 с.
14. Ткачук В.М. Алгоритми і структури даних: Навчальний посібник. — Івано-Франківськ: Видавництво Прикарпатського національного університету імені Василя Стефаника, 2016. — 286 с. (С. 227-233).
15. Хьюз Дж., Мичтом Дж. Структурний підхід к програмуванню. — Москва: Мир, 1980. — 280 с.

**Інформаційні ресурси**

1. [courses.prometheus.org.ua/courses/Prometheus/CS50/2016\\_T1/about](http://courses.prometheus.org.ua/courses/Prometheus/CS50/2016_T1/about)
2. Кампус НТУУ “КПІ” — <http://login.kpi.ua/>
3. Науково-технічна бібліотека НТУУ “КПІ” — <http://library.kpi.ua/1>.

**Навчальний контент**

**5. Методика опанування навчальної дисципліни (освітнього компонента)**

Навчальна дисципліна охоплює 4 години лекцій і 4 години лабораторних занять, передбачає виконання модульної контрольної роботи, яка проводиться на передостанньому лабораторному занятті і триває 1 академічну годину, залік проводиться на останньому лабораторному занятті і триває 2 академічні години.

На лекціях студенти одержують основний теоретичний матеріал. Основні завдання циклу лабораторних занять полягають в тому, щоб студенти закріпили матеріал лекцій і отримали практичні навички у розробці алгоритмів і проектуванні програм з використанням сучасних технологій структурного програмування, освоєння роботи на сучасних обчислювальних засобах та тестування програм.

Передбачається одна модульна контрольна робота, яка проводиться у кінці семестру. Мета контрольної роботи полягає у перевірці засвоєння матеріалу дисципліни. Для проведення контрольної роботи виділяється 2 години за рахунок лабораторних занять. На контрольну роботу виносяться: програмування арифметичних виразів, обробка одновимірних і двовимірних масивів, алгоритми сортування.

№ з/п	Назви розділів, тем, занять
<b>Розділ 1. Основи алгоритмізації</b>	
<b>Тема 1. Поняття алгоритму. Подання алгоритмів. Типові алгоритмічні структури програмування</b>	
1	Лекція 1.1. Поняття алгоритму. Подання алгоритмів. Типові алгоритмічні структури програмування Поняття алгоритму. Властивості алгоритмів. Складність алгоритмів. Способи подання алгоритмів (стор. 16-37 [1], стор. 27-36, 138-142 [3], стор. 15-18 [6], стор. 47-53 [7], стор. 29-69 [8]). Типи обчислювальних процесів: лінійні, розгалужені, циклічні (стор. 72-93 [15]).
<b>Тема 2. Позиційні і непоозиційні системи числення</b>	
2	Лекція 1.2. Позиційні і непоозиційні системи числення Позиційні і непоозиційні системи числення. Двійкова, вісімкова і шістнадцяткова системи числення. Переходи між поданнями чисел у різних системах (стор. 331-350 [5], стор. 9-15 [6]). Подання дійсних чисел у комп'ютері (стор. 9-15 [6]; [16]).
<b>Тема 3. Лінійні алгоритми</b>	
<b>Тема 4. Алгоритми з розгалуженнями</b>	
3	Лекція 1.3. Алгоритми з розгалуженнями Поняття алгоритму з розгалуженнями. Логічні вирази. Особливості обчислення логічних виразів. Розробка алгоритмів і програмування розгалужених обчислювальних процесів: послідовні умови, вкладені умови, використання оператора вибору (стор. 35-38, 44-50 [6]; стор. 40-45 [13]).
<b>Тема 5. Циклічні алгоритми</b>	
4	Лекція 1.4 Циклічні алгоритми Поняття алгоритму з циклами. Реалізація алгоритму з циклами. Особливості використання циклів з передумовою, післяумовою, з параметром. Вкладені цикли. Ітераційні циклічні процеси (стор. 50-57, 64-76 [6]; стор. 34-38 [7]; стор. 48-69 [10]; стор. 40-45 [13]).
5	Лабораторне заняття 1.1. Циклічні алгоритми

<b>Тема 6. Модульна структура програми. Рекурсивні алгоритми</b>	
6	Лекція 2.1. Модульна структура програми. Рекурсивні алгоритми Поняття модуля і модульного програмування. Модульне програмування: його переваги і недоліки. Проектування й програмування великих програм згори вниз і знизу вгору. Функції в мові програмування С (стор. 86-98 [6]; стор. 79-89 [13]; стор. 227-233 [14]; стор. 29-35, 232-238 [15]). Поняття рекурсії. Розробка рекурсивних алгоритмів. Переваги й недоліки використання рекурсії (стор. 150-169 [2]; стор. 98 [6]; стор. 79-89 [13]; стор. 227-233 [14]; стор. 29-35, 232-238 [15]).
<b>Тема 7. Алгоритми роботи з масивами</b>	
7	Лекції 2.2. Алгоритми роботи з масивами Поняття масиву; одновимірний масив. Основні типові алгоритми роботи з одновимірними масивами: знаходження мінімуму / максимуму; перестановка елементів; циклічні зсуви (стор. 14-30, 44-48 [2]; стор. 102-113 [6]). Сортування масивів (стор.199-202 [6]). Поняття двовимірного масиву. Робота з секторами матриці, обхід елементів матриці. (стор. 123-134, 149-152, 203-208 [6]).
8	Лабораторне заняття 1.2. Алгоритми роботи з масивами
<b>Тема 8. Алгоритмізація типових обчислювальних задач</b>	
9	Лабораторне заняття 1.3. <b>Модульна контрольна робота</b>
10	Лабораторне заняття 2. <b>Залік</b>

## **6. Самостійна робота студента**

*Оскільки для студентів заочної форми навчання на аудиторні заняття виділяється незначна кількість годин, то основна частка матеріалу виноситься на самостійне опрацювання. Самостійна робота студентів заочної форми навчання є невід'ємною складовою процесу навчання і основним способом оволодіння навчальним матеріалом.*

*Теми, які виносяться на самостійну роботу студентів.*

### **Тема 1. Поняття алгоритму. Подання алгоритмів. Типові алгоритмічні структури програмування**

Складність алгоритму Приклади алгоритмів і їхнього подання. Типи обчислювальних процесів: лінійні, розгалужені, циклічні. (стор. 16-37 [1], стор. 27-36, 138-142 [3], стор. 15-18 [6], стор. 47-53 [7], стор. 29-69 [8]). Типи обчислювальних процесів: лінійні, розгалужені, циклічні. Приклад розробки простого алгоритму. Різноманітність алгоритмів розв'язування задачі. Залежність алгоритму і реалізації алгоритму від використовуваної мови програмування (стор. 15-18 [6], стор. 72-93 [15]).

*Лабораторна робота 1.* Основи роботи в інтегрованому середовищі Visual Studio — С (консольний режим)

*Лабораторна робота 2.* Оформлення звітів з лабораторних робіт засобами текстового редактора Word (побудова блок-схем, вставка формул, подання результатів роботи)

### **Тема 2. Позиційні і непозиційні системи числення**

Подання чисел у позиційних системах. Двійкова, вісімкова і шістнадцяткова системи числення (стор. 9-15 [6]). Подання цілих чисел у комп'ютері. Подання дійсних чисел у комп'ютері. Особливості роботи з дійсними числами. Приклади помилок при роботі з дійсними числами. Бінарні й небінарні коди Грея, їхнє використання (стор. 331-350 [5], стор. 9-15 [6]; [16]).

*Лабораторна робота 3.* Позиційні системи числення

*Лабораторна робота 4.* Робота з плаваючою арифметикою на комп'ютері

### **Тема 3. Лінійні алгоритми**

Поняття лінійного алгоритму. Програмування виразів. Структура лінійного алгоритму. Приклади лінійних алгоритмів. Програмування виразів. Оцінка складності виразів; оптимізація обчислень (стор. 27-35 [6]; стор. 32-34 [13]).

*Лабораторна робота 5.* Програмування найпростіших обчислювальних алгоритмів лінійної структури

#### **Тема 4. Алгоритми з розгалуженнями**

Цикломатична складність алгоритму. Приклади алгоритмів з розгалуженнями. Алгоритм повного розв'язування квадратного рівняння. Оцінка обчислювальної складності алгоритмів з розгалуженнями. Особливості обчислення логічних виразів. Оптимізація алгоритмів з розгалуженнями (стор. 35-38, 44-50 [6]; стор. 40-45 [13]).

*Лабораторна робота 6.* Розробка і реалізація алгоритмів розгалужених процесів з послідовною перевіркою умов

*Лабораторна робота 7.* Розробка і реалізація алгоритмів розгалужених процесів з вкладеною перевіркою умов

#### **Тема 5. Циклічні алгоритми**

Запобігання механічним помилкам під час вводу інформації. Зациклювання роботи програми. Оригінальний алгоритм Евкліда на основі віднімання. Алгоритм Евкліда на основі ділення: Двійковий алгоритм Евкліда (алгоритм Стайна). Обчислення сум з дуже малими і великими числами. Реалізація схеми Горнера. Обчислення "машинного епсилона". Метод дотичних Ньютона обчислення кореня цілого степеня. Ітераційні алгоритми обчислення значення числа  $\pi$ . Виконання обчислень з точністю до "машинного нуля". Складність алгоритмів з циклами; оптимізація виконання, застосування мемоізації (стор. 50-57, 64-76 [6]; стор. 34-38 [7]; стор. 48-69 [10]; стор. 40-45 [13]).

*Лабораторна робота 8.* Проектування алгоритмів і програм циклічної структури. Цикли з параметром, передумовою і післяумовою.

*Лабораторна робота 9.* Проектування алгоритмів і програм циклічної структури. Вкладені цикли.

*Лабораторна робота 10.* Проектування алгоритмів і програм циклічної структури. Ітераційні алгоритми.

#### **Тема 6. Модульна структура програми. Рекурсивні алгоритми**

Модульне програмування: його переваги і недоліки. Модульна структура програми. Проектування й програмування великих програм згори вниз і знизу вгору. Функції в мові програмування C. Приклади модульних алгоритмів. Оцінка складності модульного алгоритму (стор. 86-98 [6]; стор. 79-89 [13]; стор. 227-233 [14]; стор. 29-35, 232-238 [15]). (стор. 29-35, 232-238 [15]).

*Лабораторна робота 11.* Алгоритми модульної структури

*Лабораторна робота 12.* Рекурсивні алгоритми

#### **Тема 7. Алгоритми роботи з масивами**

Поняття структури даних. Прості й інтегровані структури даних. Поняття масиву; одновимірний масив. Основні алгоритми роботи з одновимірними масивами Пошук елемента в масиві. Пошук з бар'єром. Бінарний пошук. Сортування масивів: метод бульбашки і його модифікації, шейкер-сортування, обмінне сортування простим вибором з пошуком мінімуму/максимуму, метод простих вставок, сортування Шелла. Злиття відсортованих масивів. Приклади алгоритмів з використанням одновимірних масивів. Обчислення факторіала великого числа. Решето Ератосфена (стор. 14-30, 44-48 [2]; стор. 102-113, 199-202 [6]). Поняття двовимірного масиву. Типові алгоритми роботи з двовимірними масивами. Сортування елементів матриці. Робота з секторами матриці, обхід елементів матриці. Розріджені матриці. Приклади алгоритмів з використанням двовимірних масивів. Структури. Масиви структур (стор. 123-134, 149-152, 203-208 [6]).

*Лабораторна робота 13.* Алгоритми обробки одновимірних масивів

*Лабораторна робота 14.* Алгоритми сортування масивів структур

*Лабораторна робота 15.* Алгоритми обробки двовимірних масивів

#### **Тема 8. Алгоритмізація типових обчислювальних задач**

Вибір оптимального алгоритму розв'язування задачі, шляхи оптимізації алгоритмів (на прикладі вибору алгоритму для обчислення кількості сполук). Генерація бінарних і небінарних

кодів Грея (стор. 82-99 [3]; стор. 331-349 [5]; стор. 43 [9]). Використання випадкових чисел. Алгоритми генерації рівномірно розподілених випадкових чисел: метод середин квадратів, метод серединних добутків, метод перемішування, лінійний конгруентний метод, інші методи генерації випадкових чисел (стор. 19-223 [4]). Генерація всіх підмножин множини. Генерація перестановок без повторень. Генерація перестановок з повтореннями. Генерація сполук без повторень. Генерація сполук з повтореннями. Генерація розміщень без повторень. Генерація розміщень з повтореннями (стор. с. 370-444 [5]).

*Лабораторна робота 16. Виконання додаткового завдання на вибір студента*  
*Модульна контрольна робота*  
*Залік*

## Політика та контроль

### 7. Політика навчальної дисципліни (освітнього компонента)

**Відвідування занять.** Відвідування лекційних і практичних занять є обов'язковим за винятком **поважних** причин (хвороба, непередбачувані обставини). У разі пропуску занять без поважних причин, а також через порушення граничного терміну виконання завдання студент може отримати 80% від максимальної оцінки за відповідне завдання.

**Пропущені контрольні заходи оцінювання.** Кожен студент має право відпрацювати пропущені з поважної причини заняття за рахунок самостійної роботи.

**Процедура оскарження результатів контрольних заходів оцінювання.** Студенти мають право аргументовано оскаржити результати контрольних заходів, пояснивши, з яким критерієм не погоджуються.

**Календарний контроль** проводиться з метою підвищення якості навчання студентів та моніторингу виконання студентом вимог силабусу.

**Академічна доброчесність.** Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

**Норми етичної поведінки.** Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

**Інклюзивне навчання.** Засвоєння знань та умінь в ході вивчення дисципліни «Алгоритми та структури даних-1. Основи алгоритмізації» може бути доступним для більшості осіб з особливими освітніми потребами, крім здобувачів з серйозними вадами зору, які не дають можливості виконувати завдання з використанням персональних комп'ютерів, ноутбуків та/або інших технічних засобів.

**Навчання іноземною мовою.** У ході виконання завдань студентам може бути рекомендовано звернутися до англomовних джерел.

**Призначення заохочувальних та штрафних балів** Відповідно до Положення про систему оцінювання результатів навчання сума всіх заохочувальних балів не може перевищувати 10% рейтингової шкали оцінювання.

Критерій	Заохочувальні бали		Штрафні бали	
	Ваговий бал	Критерій	Ваговий бал	Критерій
Написання тез, статті (за тематикою навчальної дисципліни)	5 балів	-	-	-
Участь у міжнародних, всеукраїнських та/або інших заходах та/або конкурсах (за тематикою навчальної дисципліни)	5 балів	-	-	-

Реалізація алгоритму, який не вивчається в курсі	1-2 бали	-	-
--	----------	---	---

Підготовка до лабораторних занять і контрольних заходів здійснюється під час самостійної роботи студентів з можливістю консультування з викладачем у визначений час консультацій або за допомогою електронних засобів (електронна пошта, телеграм, зум).

## 8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Семестрова атестація проводиться у вигляді заліку. Для оцінювання результатів навчання застосовується 100-бальна рейтингова система та університетська шкала.

**Поточний контроль:** поточні опитування, виконання і захист лабораторних робіт.

**Модульна контрольна робота:** за семестровим (кредитним) модулем передбачається одна модульна контрольна робота, яка проводиться у кінці семестру. Мета контрольної роботи полягає у перевірці засвоєння матеріалу дисципліни. Для проведення контрольної роботи виділяється 1 година за рахунок лабораторних занять. На контрольну роботу виносяться: програмування арифметичних виразів, обробка одновимірних і двовимірних масивів, алгоритми сортування

**Семестровий контроль:** залік.

**Умови допуску до семестрового контролю:** необхідною умовою допуску до заліку є виконання і захист усіх лабораторних робіт і модульної контрольної роботи, а також відпрацювання пропущених занять.

### Перелік контрольних заходів

№ п/п	Контрольний захід оцінювання	Ваговий бал	Кількість	Разом
1	Захист лабораторної роботи:			
	— першої;	1	1	1
	— другої;	2	1	2
	— решти	4	14	56
2	Звіт з лабораторної роботи	1	16	16
3	Модульна контрольна робота	25	1	25
	Разом			100

### Рейтингова система оцінювання результатів навчання

1. Оцінка з дисципліни виставляється за багатобальною системою з подальшим перерахуванням у 6-бальну.

2. Максимальна кількість балів з дисципліни дорівнює 100.

3. При нарахуванні балів за окремими видами робіт рейтинг студента складається з балів, які він отримав за:

1) виконання та захист лабораторних робіт, звіти з лабораторних робіт;

2) написання модульної контрольної роботи (МКР);

3) виконання залікової роботи (виконується з метою покращення сумарного залікового булу) контрольну роботу (ЗКР).

### Система рейтингових (вагових) балів та критерії оцінювання

#### 1) Виконання та захист лабораторних робіт, звіти з лабораторних робіт

Виконання та захист лабораторних робіт: оцінюються 16 лабораторних робіт (15 тематичних і 1 на виконання додаткового завдання за бажанням студента). Максимальний ваговий бал однієї лабораторної роботи: першої — 1 бал, другої — 2 бали, решти — по 4 бали. Максимальна кількість балів за всі лабораторні роботи (виконання і захист роботи) дорівнює 1 бал + 2 бали + 4 бали × 14 = 59 балів.



Звіти з лабораторних робіт: оцінюються 16 звітів з лабораторних робіт. Максимальний ваговий бал одного звіту з лабораторної роботи — 1. Максимальна кількість балів за всі звіти дорівнює  $1 \text{ бал} \times 16 = 16$  балів. Отже,  $r_{\text{лр}} = 59 + 16 = 75$  балів.

Критерії оцінювання.

*якість захисту лабораторної роботи:*

— студент правильно виконав лабораторну роботу і повністю виконав усі надані до захисту завдання (відповів на запитання) — 1/2/4 бали;

— студент правильно виконав лабораторну роботу і правильно виконав всі надані для захисту завдання, але допустив несуттєві неточності — 0,5/1/3 бали;

— студент виконав лабораторну роботу і частково виконав всі надані для захисту завдання — 0,5/0,5/2 бали;

— студент не повністю виконав лабораторну роботу і при виконанні завдання (у відповідях на запитання) допустив суттєві неточності — 0 балів;

*виконання звіту з лабораторної роботи:*

— звіт оформлено повністю і правильно — 1 бал;

— звіт виконано більше, ніж наполовину, але не повністю — 0,75 бала;

— звіт виконано наполовину — 0,5 бала;

— звіт виконано менше, ніж наполовину або не виконано зовсім — 0 балів;

№ розр. роб.	Тема лабораторної роботи	Кільк. балів
1	Основи роботи в інтегрованому середовищі Visual Studio — С (консольний режим)	1 + 1 = 2
2	Оформлення звітів з лабораторних робіт засобами текстового редактора Word (побудова блок-схем, вставка формул, подання результатів роботи)	2 + 1 = 3
3	Позиційні системи числення	4 + 1 = 5
4	Робота з плаваючою арифметикою на комп'ютері	4 + 1 = 5
5	Програмування найпростіших обчислювальних алгоритмів лінійної структури	4 + 1 = 5
6	Розробка і реалізація алгоритмів розгалужених процесів з послідовною перевіркою умов	4 + 1 = 5
7	Розробка і реалізація алгоритмів розгалужених процесів з вкладеною перевіркою умов	4 + 1 = 5
8	Проектування алгоритмів і програм циклічної структури. Цикли з параметром, передумовою і післяумовою	4 + 1 = 5
9	Проектування алгоритмів і програм циклічної структури. Вкладені цикли.	4 + 1 = 5
10	Проектування алгоритмів і програм циклічної структури. Ітераційні алгоритми	4 + 1 = 5
11	Алгоритми модульної структури	4 + 1 = 5
12	Рекурсивні алгоритми	4 + 1 = 5
13	Алгоритми обробки одновимірних масивів	4 + 1 = 5
14	Алгоритми сортування масивів структур	4 + 1 = 5
15	Алгоритми обробки двовимірних масивів	4 + 1 = 5
16	Виконання додаткового завдання на вибір студента	4 + 1 = 5
Усього		75

Заохочувальні та штрафні бали за:

— виконання лабораторної роботи з використанням власного оптимального алгоритму +0,5 бала;

— несвоєчасний захист лабораторної роботи або несвоєчасна здача звіту з лабораторної роботи -0,5 бала.

## 2). Написання модульної контрольної роботи (МКР)

Контрольна робота складається з 5 завдань (максимальний бал за кожне завдання — 5.

Максимальний ваговий бал —  $5 \times 5 = 25$  балів. Отже,  $r_{\text{мкр}} = 25$ .

Критерії оцінювання

*якість виконання кожного завдання оцінюється:*

- завдання виконано повністю і правильно протягом відведеного часу — 5 балів;
- завдання виконано повністю протягом відведеного часу, але має несуттєві неточності — 4 бали;
- завдання виконано більше, ніж наполовину протягом відведеного часу — 3 бали;
- завдання виконано наполовину протягом відведеного часу — 2 бали;
- завдання виконано менш ніж наполовину, але використано правильний підхід до розкриття його суті — 1 бал;
- завдання має суттєві неточності, або невиконане протягом відведеного часу — по 0 балів.

## 3). Написання залікової контрольної роботи (ЗКР)

Кожне завдання залікової роботи містить по 4 питання — 2 теоретичних і 2 практичних, кожне з яких має максимальний бал — 10. Максимальний ваговий бал —  $10 \text{ балів} \times 4 = 40$  балів. Отже,  $r_{\text{зкр}} = 40$ .

Якість відповіді на кожне питання оцінюється:

- завдання виконано повністю і правильно протягом відведеного часу — 10 балів;
- завдання виконано повністю протягом відведеного часу, але має несуттєві неточності — 8-9 балів;
- завдання виконано більше, ніж наполовину протягом відведеного часу — 6-7 балів;
- завдання виконано наполовину протягом відведеного часу — 5 балів;
- завдання виконано менш ніж наполовину, але використано правильний підхід до розкриття його суті — по 2-4 бали;
- завдання має суттєві неточності або невиконане протягом відведеного часу — 0 балів.

### **Розрахунок шкали (R) рейтингу:**

Максимальна сума вагових балів контрольних заходів протягом семестру становить:

$$R = r_{\text{лр}} + r_{\text{мкр}} = 75 + 25 = 100 \text{ балів.}$$

Таким чином, рейтингова шкала з кредитного модуля становить 100 балів.

Необхідною умовою допуску до заліку є захист лабораторних робіт і стартовий рейтинг (R) не менше  $0,4 \times 100 = 40$  балів.

Студенти, які набрали протягом семестру рейтинг з кредитного модуля  $0,4R \leq r < 0,6R$ , зобов'язані виконувати залікову контрольну роботу. Студенти, які набрали протягом семестру рейтинг  $r \geq 0,6R$  можуть підвищити свій бал на заліку.

Якщо студент виконував залікову контрольну роботу з метою покращення залікового бала, то сума вагових балів контрольних заходів становить:

$$R = r + r_{\text{зкр}} = 60 + 40 = 100 \text{ балів.}$$

Для отримання студентом відповідних оцінок (ECTS та традиційних) його рейтингова оцінка RD переводиться згідно з таблицею:

RD	Традиційна оцінка
95...100	відмінно
85...94	дуже добре
75...84	добре
65...74	задовільно
60...64	достатньо
40...59	незадовільно
$r_c < 40$	не допущений

## 9. Додаткова інформація з дисципліни (освітнього компонента)

*Перелік теоретичних питань, які виносяться на семестровий контроль (залік):*

1. Поняття алгоритму.
2. Властивості алгоритмів.
3. Складність алгоритмів.
4. Способи подання алгоритмів.
5. Приклади алгоритмів і їхнього подання.
6. Типи обчислювальних процесів: лінійні, розгалужені, циклічні.
7. Різноманітність алгоритмів розв'язування задачі.
8. Залежність алгоритму і реалізації алгоритму від використовуваної мови програмування.
9. Позиційні і непозиційні системи числення.
10. Подання чисел у позиційних системах.
11. Двійкова, вісімкова і шістнадцяткова системи числення.
12. Переходи між поданнями чисел у різних системах.
13. Подання цілих чисел у комп'ютері.
14. Подання дійсних чисел у комп'ютері.
15. Особливості роботи з дійсними числами.
16. Бінарні й небінарні коди Грея, їхнє використання.
17. Поняття лінійного алгоритму.
18. Програмування виразів.
19. Структура лінійного алгоритму.
20. Оцінка складності виразів; оптимізація обчислень.
21. Поняття алгоритму з розгалуженнями.
22. Логічні вирази.
23. Розробка алгоритмів і програмування розгалужених обчислювальних процесів: послідовні умови, вкладені умови, використання оператора вибору.
24. Цикломатична складність алгоритму.
25. Алгоритм повного розв'язування квадратного рівняння.
26. Оцінка обчислювальної складності алгоритмів з розгалуженнями.
27. Особливості обчислення логічних виразів.
28. Оптимізація алгоритмів з розгалуженнями.
29. Поняття алгоритму з циклами.
30. Реалізація алгоритму з циклами.
31. Особливості використання циклів з передумовою, післяумовою, з параметром.
32. Цикли з двома параметрами.
33. Вкладені цикли.
34. Ітераційні циклічні процеси.
35. Запобігання механічним помилкам під час вводу інформації.
36. Зациклювання роботи програми.
37. Оригінальний алгоритм Евкліда на основі віднімання.
38. Алгоритм Евкліда на основі ділення: Двійковий алгоритм Евкліда (алгоритм Стайна).
39. Обчислення сум з дуже малими і великими числами.
40. Реалізація схеми Горнера.
41. Обчислення "машинного епсилона".
42. Метод дотичних Ньютона обчислення кореня цілого степеня.
43. Ітераційні алгоритми обчислення значення числа  $\pi$ .
44. Виконання обчислень з точністю до "машинного нуля".
45. Складність алгоритмів з циклами; оптимізація виконання, застосування мемоізації.
46. Поняття модуля і модульного програмування.
47. Властивості модулів.
48. Модульне програмування: його переваги і недоліки.
49. Модульна структура програми.
50. Проектування й програмування великих програм згори вниз і знизу вгору.
51. Функції в мові програмування C.
52. Оцінка складності модульного алгоритму.
53. Поняття рекурсії.
54. Розробка рекурсивних алгоритмів.
55. Рекурсивні описи.
56. Рекурсивне звернення.
57. Пряма й непряма рекурсія.
58. Рекурсивний спуск, підйом, глибина рекурсії.

59. Переваги й недоліки використання рекурсії.
60. Оцінка складності рекурсивного алгоритму.
61. Поняття структури даних.
62. Прості й інтегровані структури даних.
63. Поняття масиву; одновимірний масив.
64. Основні типові алгоритми роботи з одновимірними масивами: знаходження мінімуму / максимуму; перестановка елементів; циклічні зсуви.
65. Пошук елемента в масиві.
66. Пошук з бар'єром.
67. Бінарний пошук.
68. Сортування масивів: метод бульбашки і його модифікації.
69. Сортування масивів: шейкер-сортування.
70. Сортування масивів: обмінне сортування простим вибором з пошуком мінімуму.
71. Сортування масивів: обмінне сортування простим вибором з пошуком максимуму.
72. Сортування масивів: метод простих вставок.
73. Сортування масивів: сортування Шелла.
74. Злиття відсортованих масивів.
75. Обчислення факторіала великого числа.
76. Решето Ератосфена.
77. Поняття двовимірного масиву.
78. Типові алгоритми роботи з двовимірними масивами.
79. Сортування елементів матриці.
80. Робота з секторами матриці, обхід елементів матриці.
81. Розріджені матриці.
82. Генерація бінарних і небінарних кодів Грея.
83. Використання випадкових чисел.
84. Алгоритми генерації рівномірно розподілених випадкових чисел: метод середин квадратів.
85. Алгоритми генерації рівномірно розподілених випадкових чисел: метод серединних добутків.
86. Алгоритми генерації рівномірно розподілених випадкових чисел: метод перемішування.
87. Алгоритми генерації рівномірно розподілених випадкових чисел: вибір параметрів лінійного конгруентного методу генерації випадкових чисел.
88. Генерація всіх підмножин множини.
89. Генерація перестановок без повторень.
90. Генерація перестановок з повтореннями.
91. Генерація сполук без повторень.
92. Генерація сполук з повтореннями.
93. Генерація розміщень без повторень.
94. Генерація розміщень з повтореннями.

*Виставлення оцінки за контрольні заходи шляхом перенесення результатів проходження онлайн-курсів чи інших курсів не передбачено.*

#### **Робочу програму навчальної дисципліни (силабус):**

*Складено: доцент кафедри автоматизації проектування енергетичних процесів і систем, канд. техн .наук, доцент, Кублій Лариса Іванівна*

**Ухвалено кафедрою АПЕПС (протокол № 16 від 18.06.21)**

**Погоджено Методичною комісією ТЄФ КПІ ім. Ігоря Сікорського (протокол № 11 від 24.06.21)**