



АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ-1. БАЗОВІ КОНЦЕПЦІЇ ПРОГРАМУВАННЯ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Освітня програма	<i>Цифрові технології в енергетиці</i>
Статус дисципліни	<i>Нормативна. Цикл професійної підготовки</i>
Форма навчання	<i>Заочна</i>
Рік підготовки, семестр	<i>1 курс, осінній семестр</i>
Обсяг дисципліни	<i>150 годин / 5 кредитів ЄКТС (лекції – 4 год., лабораторні заняття – 2 год., СРС – 144 год.)</i>
Семестровий контроль/ контрольні заходи	<i>Залік, МКР</i>
Розклад занять	<i>http://roz.kpi.ua/</i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лекції і лабораторні заняття проводить: канд. техн. наук, доцент, Кублій Лариса Іванівна kublii_l_i@ukr.net, тел. 063-71-91-231 (+Телеграм), 097-558-27-17</i>
Розміщення курсу	<i>Кампус</i>

Програма навчальної дисципліни

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

Знання мов програмування, знання і застосування існуючих і розробка нових алгоритмів є невід'ємною складовою роботи кожного програміста. Згідно з робочим навчальним планом кредитний модуль "Алгоритмізація та програмування-1. Базові концепції програмування" (ПО-6.1) дисципліни "Алгоритмізація та програмування" викладається студентам першого року підготовки першого (бакалаврського) рівня вищої освіти ступеня "бакалавр" спеціальності "Комп'ютерні науки" освітньої програми "Цифрові технології в енергетиці" у першому навчальному семестрі. У рамках вивчення матеріалу модуля базовими є концепції структурного і модульного програмування. Студенти вивчають мову програмування C, знайомляться з системами числення, поданням числових даних у комп'ютері і основні алгоритми — класичні і сучасні, лінійні, з розгалуженнями, ітераційні і рекурсивні — зокрема, алгоритм Евкліда знаходження найбільшого спільного дільника, решето Ератосфена пошуку простих чисел, схему Ньютона для обчислення коренів, прості алгоритми сортування елементів масивів, алгоритми, які базуються на використанні побітових операцій, та багато інших, значну увагу приділено врахуванню можливості арифметичного переповнення при роботі з цілими числами і точності обчислень при роботі з дійсними числами.

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 вчитися і оволодівати сучасними знаннями (ЗК 6);
- здатність генерувати нові ідеї (креативність) (ЗК 8);
- здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК 12);

— здатність до логічного мислення, 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 (ФК 3);

— здатність до системного мислення, застосування методології системного аналізу для дослідження складних проблем різної природи, методів формалізації та розв'язування системних задач, що мають суперечливі цілі, невизначеності та ризику (ФК 6);

— 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 (ФК 8);

— використання принципів структурного програмування, основних структур даних під час реалізації алгоритмів професійних завдань;

— володіння алгоритмічним мисленням, методами програмної інженерії для реалізації програмного забезпечення з урахуванням вимог до його якості, надійності, виробничих характеристик.

Предметом навчальної дисципліни є вивчення мови програмування C, простих структур даних і типових алгоритмів роботи з ними.

Студенти після засвоєння кредитного модуля мають продемонструвати такі результати навчання:

Знання:

— основ організації обчислювального процесу на ЕОМ;

— систем числення;

— теоретичних основ алгоритмізації, проектування та тестування програм;

— методів структурного програмування;

— алгоритмів розв'язування типових задач;

— засобів програмування алгоритмічної мови C;

вміння:

— застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук (ПР 1);

— проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій (ПР 5);

— 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 (ПР 9).

— застосовувати мови програмування (зокрема мову програмування C);

— проектувати компоненти програмного забезпечення;

— проектувати людино-машинний інтерфейс інформаційних систем;

— розробляти оптимальні алгоритми для широкого кола задач;

— реалізувати алгоритми мовою програмування C як закінченого програмного продукту;

— виконувати аналіз коректності програм, їхнє налагодження та тестування з використанням сучасних технологій програмування;

— застосовувати набуті базові знання з дисципліни в професійній діяльності під час розробки, налагодження та експлуатації ІС та технологій;

— застосовувати принципи структурного програмування при проектуванні і розробці програм;

— правильно вибирати й застосовувати основні структури даних під час програмної реалізації алгоритмів;

— аналізувати та програмувати виключні ситуації у програмних продуктах;

— застосовувати типові алгоритми обробки даних.

Додаткові знання:

— особливостей предметного середовища, застосування відповідних структур даних, спеціалізованих алгоритмів, різних мовних конструкцій, 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 "Алгоритмізація та програмування-1. Базові концепції програмування" не має дисциплін, які її забезпечують, вона вивчається на основі знань, отриманих у середній загальноосвітній школі. Компетенції, отримані студентами в процесі вивчення цього модуля, застосовуються при вивченні навчальних дисциплін: "Алгоритмізація та програмування-2. Процедурне програмування", "Проектування та аналіз обчислювальних алгоритмів", "Програмування алгоритмічних структур", "Об'єктно-орієнтоване програмування", "Операційні системи", "Безпека інформаційних систем" "Технології розробки програмного забезпечення".

3. Зміст навчальної дисципліни

Розділ 1. Організація програм

Тема 1.1. Архітектура комп'ютерів, принципи фон Неймана

Тема 1.2. Позиційні системи числення

Тема 1.3. Поняття алгоритму та типові алгоритмічні структури програмування

Тема 1.4. Елементи алгоритмічних мов: концепція типів даних, імена, значення, вказівники, змінні, константи, операції, вирази

Тема 1.5. Структурне програмування: послідовність, розгалуження та цикли

Тема 1.6. Методології розробки програм: низхідне та висхідне проектування, модульне програмування

Тема 1.7. Процедурно-орієнтоване програмування. Рекурсія

Розділ 2. Структури даних і алгоритми

Тема 2.1. Організація даних (масиви, рядки, структури) та алгоритми їхнього оброблення

Тема 2.2. Алгоритмізація типових обчислювальних задач

4. Навчальні матеріали та ресурси

Базова література

- Кублій Л.І. Алгоритми та структури даних. Основи алгоритмізації [Електронний ресурс]: підручник для здобувачів ступеня бакалавра за спеціальністю 121 "Інженерія програмного забезпечення" / Л.І. Кублій; КПІ ім. Ігоря Сікорського. — Електронні текстові дані (1 файл: 21,3 Мбайт). — Київ: КПІ ім. Ігоря Сікорського, 2022. — 528 с. — <https://ela.kpi.ua/handle/123456789/48282>.
- 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
- Kernighan Brian W., Ritchie Dennis M. The C Programming Language: Second edition. — USA: Prentice Hall, 1988. — 288 p. — http://cslabcms.nju.edu.cn/problem_solving/images/c/cc/The_C_Programming_Language_%282nd_Edition_Ritchie_Kernighan%29.pdf (Керніган Б., Річі Д. Мова програмування С. Пер. з англ., 2-е вид. — 2013. — 232 с. — <http://programming.in.ua/programming/c-language/227-book-programming-c-kernighan.html>)
- Шпак З.Я. Програмування мовою С. — Львів: Оріяна-Нова, 2006. — 432 с. — http://pdf.lib.vntu.edu.ua/books/2017/Shpak_2006_432.pdf

Додаткова література

- Войтенко В.В., Морозов А.В. С/С++: Теорія та практика: навч.-метод. Посібник. 2-е вид. — Житомир: ЖДТУ, 2004. — 325 с. — <http://programming.in.ua/programming/c-language/183-c-or-c-book-voitenko-and-morozov.html>
- Гарасимчук О. І. Генератори псевдовипадкових чисел, їх застосування, класифікація, основні методи побудови і оцінка якості [Електронний ресурс] / О. І. Гарасимчук, В. М. Максимович // Науково-технічний журнал «Захист інформації». — 2003. — № 30. — С. 29-36. — Режим доступу: <http://jrn1.nau.edu.ua/index.php/ZI/article/viewFile/4270/4405> (дата звернення: 12.02.2022).
- 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 .
- Fog Agner. 4. Instruction tables [Electronic resource]. — Technical University of Denmark, 22.03.2021. — 424 p. — https://www.agner.org/optimize/instruction_tables.pdf (date of access: 23.12.2021).
- IEEE Standard for Floating-Point Arithmetic. — New York, 2008. — 58 p.

Інформаційні ресурси

- Кампус НТУУ "КПІ" — <http://login.kpi.ua/>
- Науково-технічна бібліотека НТУУ "КПІ імені Ігоря Сікорського" — <https://www.library.kpi.ua/>

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчальна дисципліна охоплює 4 години лекцій і 2 години лабораторних занять, передбачає виконання модульної контрольної роботи, яка проводиться на лабораторному занятті і

триває 1 академічну годину, залік виставляється на основі набраних за виконання й захист лабораторних робіт балів. 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 програмування, освоєння роботи на сучасних обчислювальних засобах та тестування програм.

За семестровим (кредитним) модулем передбачається одна модульна контрольна робота, яка проводиться в кінці семестру. Мета модульної контрольної роботи полягає в перевірці засвоєння матеріалу дисципліни. Контрольна робота містить 5 завдань. На контрольну роботу виносяться такі теми: програмування арифметичних виразів, програмування умовних операторів, обробка одновимірних і двовимірних масивів, алгоритми сортування.

№	Назви розділів, тем, занять
Розділ 1. Організація програм	
Тема 1.1. Архітектура комп'ютерів, принципи фон Неймана	
Тема 1.2. Позиційні і непозиційні системи числення	
1	<u>Лекція 1.1. Структура ПК. Позиційні системи числення</u> Структура ПК. Принципи фон Неймана. (стор. 63 [1]; [8]). Двійкова, вісімкова, шіснадцяткова системи числення. Подання даних в ЕОМ. Особливості роботи з дійсними числами. стор. 61-112 [1], стор. 9-15 [2]).
Тема 1.3. Поняття алгоритму та типові алгоритмічні структури програмування	
Тема 1.4. Елементи алгоритмічних мов: концепція типів даних, імена, значення, вказівники, змінні, константи, операції, вирази	
Тема 1.5. Базові структури програмування: послідовність, розгалуження, цикли	
2	<u>Лекція 1.2. Алгоритми і алгоритмічні структури</u> Засоби подання алгоритмів. Лінійні, розгалужені та циклічні обчислювальні процеси. (стор. 9-36 [1], стор. 15-18 [2], стор. 27-30 [5], стор. 19-40 [7]). Структура програми мовою С. Типи даних. Вирази, перетворення типів операндів у виразах. Операції. Стиль запису алгоритмів мовою С. (стор. 113-135 [1], стор. 18-20, 27-31, 181-191 [2], стор. 37-60 [3-укр], стор. 11-64 [4], стор. 6-14, 19-26 [5]). Логічні вирази. Умовний оператор. (стор. 143-163 [1], стор. 35-50 [2], стор. 61-67 [3], 87-94 [4], стор. 26, 32-36 [5]). Оператор циклу з передумовою. Оператор циклу з післяумовою (стор. 167-222 [1], стор. 52-57 [2], 67-73 [3], стор. 100-105 [4], стор. 36-39 [5]). Оператор циклу з параметром (стор. 168-218 [1], стор. 50-52 [2], стор. 67-73 [3], стор. 94-105 [4], стор. 39-42 [5]).
3	Лабораторне заняття 1.1. Циклічні алгоритми
Тема 1.6. Процедурно-орієнтоване програмування. Рекурсія	
4	<u>Лекція 2.1. Модульне програмування. Рекурсія</u> Модульне програмування. Формальні й фактичні параметри. Передача параметрів за значенням і за посиланням. Локальні й глобальні змінні. Статичні змінні (стор. 229-270 [1], стор. 86-98 [2], стор. 75-96, 107-110 [3], стор. 183-194 [4], стор. 99-119 [5]). Рекурсія. Розробка рекурсивних алгоритмів (стор. 247-266 [1], стор. 98 [2], стор. 96-98 [3], стор. 222-227 [4], стор. 110-112 [5]).
Розділ 2. Структури даних і алгоритми	
Тема 2.1. Організація даних (масиви, рядки, структури) та алгоритми їх оброблення	
Тема 2.3. Алгоритмізація типових обчислювальних задач	
5	<u>Лекція 2.2. Одновимірні й двовимірні масиви. Рядки символів. Структури</u> Поняття масиву. Оголошення масивів. Можливості роботи з масивами. Передача параметрів-масивів. Сортування елементів масиву (стор. 273-341 [1], 102-113, 199-202 [2], стор. 110-114 [3], стор. 125-131, 195-197 [4], стор. 61-66, 76-80 [5]). Двовимірні масиви. Можливості роботи з двовимірними масивами (стор. 352-387 [1], стор. 123-134, 203-208 [2], 126-128 [3], стор. 131-140 [4], стор. 66-69 [5]). Символьні дані. Рядки символів. Функції роботи з рядками (стор. 369-373 [1], стор. 138-149 [2], стор. 52-54 [3], стор. 48-51, 141-162 [4], стор. 71-76 [5]). Структури. Доступ до членів структури (стор. 321-335 [1], стор. 149-152 [2], стор. 145-164 [3], стор. 164-175 [4], стор. 80-87 [5]). Вибір алгоритму розв'язування задачі.

6	Лабораторне заняття 1.2. Алгоритми роботи з масивами
7	Лабораторне заняття 1.3. Модульна контрольна робота
8	Залік

6. Самостійна робота студента

Оскільки для студентів заочної форми навчання на аудиторні заняття виділяється незначна кількість годин, то основна частка матеріалу виносить 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.

Розділ 1. Організація програм

Тема 1.1. Архітектура комп'ютерів, принципи фон Неймана

Значення обчислювальної техніки в інженерній роботі. Коротка історична довідка. Структура ПК. Принципи фон Неймана. (стор. 63 [1]; [8]).

Лабораторна робота 1. Основи роботи в інтегрованому середовищі Visual Studio (консольний режим C++). Приклад найпростішої програми мовою C.

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

Тема 1.2. Позиційні і непозиційні системи числення

Двійкова, вісімкова, шіснадцяткова системи числення. Подання даних в ЕОМ. Особливості роботи з дійсними числами. Коды Грея (стор. 61-112 [1], стор. 9-15 [2]). Подання цілих і дробових чисел в двійковій системі числення (стор. 78-94 [1], стор. 9-15 [2], стор. 1-52 [6], [9]).

Лабораторна робота 3. Позиційні системи числення. Робота з дійсними числами на комп'ютері.

Тема 1.3. Поняття алгоритму та типові алгоритмічні структури програмування

Алгоритм та його властивості. Засоби подання алгоритмів. Правило виконання схем. Типи обчислювальних процесів. Лінійні, розгалужені та циклічні обчислювальні процеси. Приклади алгоритмів і їхнього подання. Характеристика алгоритмічних мов програмування (стор. 9-36 [1], стор. 15-18 [2], стор. 27-30 [5], стор. 19-40 [7]).

Тема 1.4. Елементи алгоритмічних мов: концепція типів даних, імена, значення, вказівники, змінні, константи, операції, вирази

Основні характеристики алгоритмічної мови C. Структура програми мовою C. Алфавіт мови. Лексичні одиниці мови C: ключові слова, ідентифікатори, константи, мітки, знаки операцій, коментарі, роздільники. Змінні, ініціалізація змінних. Іменовані константи. Типи даних. Вирази, перетворення типів операндів у виразах. Операції. Стиль запису алгоритмів мовою C (стор. 113-135 [1], стор. 18-20, 27-31, 181-191 [2], стор. 37-60 [3-укр], стор. 11-64 [4], стор. 6-14, 19-26 [5]).

Лабораторна робота 4. Програмування найпростіших обчислювальних алгоритмів лінійної структури. Використання стандартних і визначених користувачем типів даних. Введення-виведення даних

Тема 1.5. Базові структури програмування: послідовність, розгалуження, цикли

Основні структури програмування. Розробка алгоритмів і написання програм лінійних обчислювальних процесів. Оператори мови C. Оператор присвоювання. Оператор виклику функції. Блок операторів. Порожній оператор. Оператор безумовного переходу goto. Введення-виведення даних. Використання формату даних. (стор. 113-139 [1], стор. 32-35, 192-198 [2], стор. 50-55, 173-190 [3], стор. 66-84, 86-87 [4], стор. 14-27, 42-43 [5]; [8]). Оптимізація лінійних обчислювальних процесів (стор. 133-139 [1], стор. 7-13 [5]). Логічні вирази. Розробка алгоритмів і програмування розгалужених обчислювальних процесів. Умовний оператор. Вкладені умовні оператори. Оператор вибору. (стор. 143-163 [1], стор. 35-50 [2], стор. 61-67 [3], 87-94 [4], стор. 26, 32-36 [5]). Оптимізація розгалужених алгоритмів. (стор. 157-163 [1]). Поняття циклу. Розробка алгоритму й програмування циклічного процесу. Оператор циклу з передумовою. Оператор циклу з післяумовою. (стор. 167-222 [1], стор. 52-57 [2], 67-73 [3], стор. 100-105 [4], стор. 36-39 [5]). Складність і оптимізація циклічних алгоритмів. (стор. 218-222 [1], стор. 19 [5]). Циклічна структура з заданою кількістю повторень. Оператор циклу з параметром. Часткові нестандартні випадки використання циклу for. Оператори break і continue. (стор. 168-218 [1], стор. 50-52 [2], стор. 67-73 [3], стор. 94-105 [4], стор. 39-42 [5]). Розробка алгоритму й програмування ітераційних циклів. Вкладені цикли. (стор. 168-218 [1], стор. 64-76 [2], стор. 67-73 [3], стор. 100-102 [4]).

Лабораторна робота 5. Розробка і реалізація алгоритмів розгалужених процесів з послідовною перевіркою умов.

Лабораторна робота 6. Розробка і реалізація алгоритмів розгалужених процесів з вкладеною перевіркою умов (умовний оператор і оператор вибору).

Лабораторна робота 7. Проектування алгоритмів і програм циклічної структури. Цикли різних типів.

Лабораторна робота 8. Проектування алгоритмів і програм циклічної структури. Вкладені цикли, робота з дійсними числами.

Лабораторна робота 9. Проектування алгоритмів і програм циклічної структури. Ітераційні алгоритми.

Тема 1.6. Процедурно-орієнтоване програмування. Рекурсія

Методи проектування програм: висхідне і низхідне проектування і програмування. Метод проектування, покрокове уточнення. Модульне програмування. Алгоритми модульної структури. Функції. Оголошення (прототипи функцій) і опис функції. Порожній тип void. Звернення до функцій. Оператор return. Функції без параметрів. Бібліотеки функцій. Формальні й фактичні параметри. Передача параметрів за значенням і за посиланням. Локальні й глобальні змінні. Статичні змінні (стор. 229-270 [1], стор. 86-98 [2], стор. 75-96, 107-110 [3], стор. 183-194 [4], стор. 99-119 [5]). Складність і оптимізація алгоритмів модульної структури (стор. 266-270 [1]). Рекурсія. Розробка рекурсивних алгоритмів. Рекурсивні описи. Рекурсивне звернення. Приклади рекурсивних функцій. (стор. 247-266 [1], стор. 98 [2], стор. 96-98 [3], стор. 222-227 [4], стор. 110-112 [5]).

Лабораторна робота 10. Розробка програм модульної структури.

Лабораторна робота 11. Рекурсивні алгоритми.

Розділ 2. Структури даних і алгоритми

Тема 2.1. Організація даних (масиви, рядки, структури) та алгоритми їх оброблення

Поняття масиву. Оголошення масивів. Можливості роботи з масивами. Передача параметрів-масивів. Введення та виведення масивів. Розмірність масивів. Розміщення у пам'яті. Тип індексів масиву. Ініціювання масивів. Одновимірні масиви. Типові алгоритми роботи з одновимірними масивами. Алгоритми обробки одновимірних масивів. Пошук елемента в масиві. Максимальний та мінімальний елементи. Сортування елементів масиву. Злиття масивів. (стор. 273-341 [1], 102-113, 199-202 [2], стор. 110-114 [3], стор. 125-131, 195-197 [4], стор. 61-66, 76-80 [5]). Складність і оптимізація алгоритмів обробки одновимірних масивів (стор. 297-320, 335-341 [1]). Багатовимірні масиви. Двовимірні масиви. Можливості роботи з двовимірними масивами. Вивід матриці. Передача багатовимірних параметрів-масивів. Типові алгоритми роботи з двовимірними масивами: визначення діагоналей, секторів, перевірка на впорядкованість даних, циклічний зсув, вставка, вилучення рядків, стовпчиків, впорядкування елементів. Розріджені матриці. Алгоритми роботи з розрідженими матрицями. (стор. 352-387 [1], стор. 123-134, 203-208 [2], 126-128 [3], стор. 131-140 [4], стор. 66-69 [5]). Складність і оптимізація алгоритмів обробки двовимірних масивів (стор. 385-387 [1]). Символьні дані. Рядки символів. Рядкова константа. Ініціалізація рядка символів. Введення-виведення рядка символів. Функції роботи з рядками. Побітові операції. Побітові зсуви. Множення і ділення на 2. Побітові логічні операції. (стор. 369-373 [1], стор. 138-149 [2], стор. 52-54 [3], стор. 48-51, 141-162 [4], стор. 71-76 [5]). Структури. Доступ до членів структури. Присвоювання структур. Масиви структур. Передача структур і членів структур функціям. Масиви і структури всередині структур. Бітові поля. (стор. 321-335 [1], стор. 149-152 [2], стор. 145-164 [3], стор. 164-175 [4], стор. 80-87 [5]).

Лабораторна робота 12. Обробка одновимірних масивів.

Лабораторна робота 13. Сортування в одновимірних масивах.

Лабораторна робота 14. Обробка двовимірних масивів. Робота з секторами матриці.

Лабораторна робота 15. Робота з рядками символів.

Лабораторна робота 16. Робота зі структурами

Тема 2.2. Алгоритмізація типових обчислювальних задач

Вибір алгоритму розв'язування задач. Генерація псевдовипадкових чисел. Генерація кодів Грея. Генерація деяких комбінаторних множин (стор. 392-501 [1]; [6]).

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 силябусу.

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 б-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. 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 $1 \text{ бал} + 2 \text{ бали} + 4 \text{ бали} \times 14 = 59 \text{ балів}$.

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 $1 \text{ бал} \times 16 = 16 \text{ балів}$. Отже, $r_{\text{лр}} = 59 + 16 = 75 \text{ балів}$.

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;

№ лаб. роб.	Тема лабораторної роботи	Кільк. балів
1	Основи роботи в інтегрованому середовищі Visual Studio (консольний режим C++). Приклад найпростішої програми мовою С.	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. Робота з дійсними числами на комп'ютері	4 + 1 = 5
4	Програмування найпростіших обчислювальних алгоритмів лінійної структури. Використання стандартних і визначених користувачем типів даних. Введення-виведення даних.	4 + 1 = 5
5	Розробка і реалізація алгоритмів розгалужених процесів з послідовною перевіркою умов.	4 + 1 = 5
6	Розробка і реалізація алгоритмів розгалужених процесів з вкладеною перевіркою умов (умовний оператор і оператор вибору).	4 + 1 = 5
7	Проектування алгоритмів і програм циклічної структури. Цикли різних типів.	4 + 1 = 5
8	Проектування алгоритмів і програм циклічної структури. Вкладені цикли, робота з дійсними числами.	4 + 1 = 5
9	Проектування алгоритмів і програм циклічної структури. Ітераційні алгоритми.	4 + 1 = 5
10	Розробка програм модульної структури.	4 + 1 = 5

11	Рекурсивні алгоритми.	4 + 1 = 5
12	Обробка одновимірних масивів.	4 + 1 = 5
13	Сортування в одновимірних масивах.	4 + 1 = 5
14	Обробка двовимірних масивів. Робота з секторами матриці.	4 + 1 = 5
15	Робота з рядками символів.	4 + 1 = 5
16	Робота зі структурами	4 + 1 = 5
<i>Усього</i>		<i>75</i>

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 — 4. Максимальний ваговий бал — $4 \times 5 = 20$ балів. Отже, $r_{\text{мкр}} = 20$.

Критерії оцінювання:

якість виконання кожного завдання оцінюється:

- завдання виконано повністю і правильно протягом відведеного часу — 4 бали;
- завдання виконано повністю протягом відведеного часу, але має несуттєві неточності — 3 бали;
- завдання виконано більше, ніж наполовину протягом відведеного часу — 2 бали;
- завдання виконано наполовину протягом відведеного часу — 1 бал;
- завдання виконано менш ніж наполовину, або 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:

$$R = r_{\text{лр}} + r_{\text{лек}} + r_{\text{мкр}} = 75 + 5 + 20 = 100 \text{ балів.}$$

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. Структура й принципи роботи ПЕОМ.
2. Поняття алгоритму. Властивості алгоритму.
3. Подання алгоритмів. Словесне подання алгоритму (метамова). Подання алгоритму за допомогою блок-схеми.
4. Типи обчислювальних структур: лінійні, розгалужені, циклічні.
5. Основні поняття програмування. Синтаксис, синтаксичні помилки. Семантика, семантичні помилки. Налагодження й тестування програм.
6. Методи трансляції програм: інтерпретація, компіляція.
7. Основні положення структурного програмування.
8. Модульне програмування. Переваги й недоліки.
9. Проектування й програмування згори вниз і знизу вгору.
10. Структурне програмування і коректність програм.
11. Системи числення. Двійкове, вісімкове, шістнадцяткове подання цілих і дробових чисел.
12. Особливості роботи з дійсними числами в комп'ютері.
13. Структура програми мовою C: декларативна й виконувана частини.
14. Лексичні одиниці мови C: ключові слова, ідентифікатори, константи.
15. Типи даних в мові C. Перетворення типів у виразаї і при присвоюванні.
16. Введення й виведення даних (функції printf, scanf). Формати.
17. Запис цілих, дійсних, логічних, символічних констант мови C.
18. Оголошення й використання констант у мові C.
19. Змінні стандартних типів мови C: цілі, дійсні, символічні.
20. Операції мови C. Операція присвоювання.
21. Операції мови C. Операція кома.
22. Арифметичні вирази.
23. Оцінка складності виразів; оптимізація обчислень.
24. Операції порівняння. Логічні вирази.
25. Умовний оператор мови C. Вкладені умовні оператори.
26. Тернарна операція ?:
27. Оператор вибору мови C.
28. Оцінка обчислювальної складності алгоритмів з розгалуженнями.
29. Поняття циклу. Оператори циклу мови C. Вкладені цикли.
30. Оператор циклу з параметром. Схема виконання.
31. Нестандартні можливості використання оператора циклу з параметром.
32. Оператор циклу з передумовою. Схема виконання.
33. Оператор циклу з післяумовою. Схема виконання.
34. Складність алгоритмів з циклами; оптимізація виконання, застосування мемоізації.
35. Функції мови C: стандартні й описувані.
36. Функції мови C: з параметрами і без параметрів.
37. Функції, які повертають значення. Оператор return.
38. Локальні, глобальні й статичні змінні.
39. Фактичні й формальні параметри.
40. Механізми передачі параметрів за адресою і за значенням.
41. Оцінка складності модульного алгоритму.
42. Рекурсивні функції. Оцінка складності рекурсивного алгоритму.
43. Одновимірні масиви мови C: опис, ініціалізація, введення-виведення.
44. Методи сортування елементів масиву: обмінне сортування простою вибіркою з пошуком мінімуму чи максимуму.
45. Методи сортування елементів масиву: обмінне сортування (метод бульбашки) з індикатором перестановки.
46. Методи сортування елементів масиву: шейкер-сортування.
47. Методи сортування елементів масиву: сортування простими вставками.
48. Двовимірні масиви мови C: опис, ініціалізація, введення-виведення.
49. Передача параметрів-масивів у функції.

50. Типові алгоритми роботи з двовимірними масивами.
51. Сортування елементів матриці.
52. Робота з секторами матриці, обхід елементів матриці.
53. Розріджені матриці.
54. Змінні символного типу і рядки символів.
55. Основні функції роботи з рядками символів.
56. Перераховувані дані: опис, ініціалізація, використання.
57. Побітові операції.
58. Структури: опис, робота, передача в функції.
59. Масиви структур: опис, робота, передача в функції.
60. Використання випадкових чисел.

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

Робочу програму навчальної дисципліни (силабус)

складено: доцент кафедри цифрових технологій в енергетиці,
канд. техн .наук, доцент, Кублій Лариса Іванівна

Ухвалено кафедрою ЦТЕ (протокол № 1 від 01.07.2022р)

Погоджено Методичною комісією факультету (протокол № 10 від 04.07.2022 р.)