Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра обчислювальної техніки

Алгоритми та структури даних. Частина 1. Основи алгоритмізації

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни	
Рівень вищої освіти	***Перший (бакалаврський)***
Галузь знань	*F Інформаційні технології*
Спеціальність	*F2 Інженерія програмного забезпечення*
Освітня програма	*Інженерія програмного забезпечення комп'ютерних систем*
Статус дисципліни	*Нормативна*
Форма навчання	*очна*
Рік підготовки, семестр	*1 курс, осінній*
Обсяг дисципліни	*4 кредитів/120 годин, денна форма: лекцій 30 годин, лабораторних робіт 14 годин та СРС 76 годин.*
Семестровий контроль/ контрольні заходи	*Залік/ Лабораторні роботи*
Розклад занять	*Згідно розкладу на осінній семестр поточного навчального року за адресою http://shedule.kpi.ua*
Мова викладання	*Українська*
Інформація про керівника курсу / викладачів	**Лектор** д.т.н, професор Анатолій Михайлович Сергієнко , *email Sergiyenko.Anatoliy@lll.kpi.ua* **Викладач** з лабораторних занять д. філ. : *Череватенко Олексій Володимирович, email chereva@ukr.net*
Розміщення курсу	*На платформі дистанційного навчання «Сікорський»: https://classroom.google.com/c/NDIyNTg2NDY3NTY4?cjc=whbtqcx*

Програма навчальної дисципліни

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

Викладання алгоритмів та структур даних в університеті має такі аспекти:

Академічний аспект. Алгоритми та структури даних - це наука, яка дає фундаментальні знання. Мета - ознайомити студентів з основними поняттями та методами представлення даних та виконання з ними алгоритмів, вимірюючи складність алгоритму.

Професійний аспект. Знання структур даних та алгоритмів є обов'язковими для інженерів та програмістів, які розробляють програмне забезпечення для комп'ютера. Мета цього курсу – сформувати навички використання мови С для програмування базових алгоритмів, які використовуються у більшості програмного забезпечення універсальних та спеціалізованих систем.

Інтелектуальні та освітні аспекти. Вивчення структур даних та алгоритмів сприяє розвитку когнітивних навичок, що є суттєвим інтелектуальним фактором у процесі створення програмного забезпечення.

В результаті вивчення цієї дисципліни досягаються такі результати навчання.

Компетенції:

ЗК 01 здатність до абстрактного мислення, аналізу та синтезу.

ЗК 02 здатність до навчання та набуття сучасних знань.

ЗК 05 Здатність вчитися і оволодівати сучасними знаннями.

ФК 01 Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

ФК 02 Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

ФК 05 Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу

ФК 14 Здатність до алгоритмічного та логічного мислення.

У відповідності до вищезазначеного, підсилені загальні та фахові компетенції дадуть наступні **знання:**

ПРН 02 Знати кодекс професійної етики, розуміти соціальну значимість та культурні аспекти інженерії програмного забезпечення і дотримуватись їх в професійній діяльності.

ПРН 03 Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.

ПРН 04 Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.

ПРН 07 Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.

ПРН 13 Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПРН 23 Вміти документувати та презентувати результати розробки програмного забезпечення.

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

Передумови, тобто знання, вивчення яких має передувати вивченню цієї дисципліни:

- базовий рівень володіння англійською мовою не нижче А1.
- базовий рівень шкільної освіти за напрямками математики, інформатики, української мови.

Знання та навички, отримані під час вивчення дисципліни «Алгоритми та структури даних», можуть бути застосовані під час вирішення навчальних та лабораторних завдань, що виникають під час засвоєння дисциплін:

- алгоритми та структури даних 2. Структури даних;
- архітектура комп’ютера;
- бази даних;

• розробка програм;
• системне програмне забезпечення;
• чисельні методи;
• комп’ютерні мережі;
• штучний інтелект.

3. Зміст навчальної дисципліни

Тема 1: Основи мови С.

Предмет і мета предмета «структури даних та алгоритми». Поняття алгоритму. Архітектура з точки зору програміста. Історія мови С. Модель мови С. Об'єкти мови С. Операції та оператори мови С. Перетворення типів у виразах.

Тема 2: Алгоритми та програми.

Конструкції для проектування алгоритму. Блокові конструкції. Конструкції управління: конструкції розгалуження, повторення, перехід. Подання алгоритмів та оператори. Діаграми діяльності. Алгоритмічні блок -схеми. Основи структурного програмування.

Основи оцінки алгоритму. Часова складність. Просторова складність. Оцінка порядку складності. Ітеративні алгоритми. Типи алгоритмів: поліноміальні комплексні та NP-повні алгоритми. Вимірювання складності в програмі. Оптимізація алгоритму. Найпростіші алгоритмічні прийоми. Метод динамічного програмування.

Підпрограми. Функції та процедури. Контекст підпрограми, інтерфейс та реалізація. Реалізація підпрограми в архітектурі. Методи передачі параметрів. Структура програми С. Програмування функцій у С. Методи передачі параметрів у підпрограму.

Тема 3: Структури даних.

Концепція абстрактного типу даних. Функції над структурами даних. Класифікації структури даних. Статичні та динамічні структури даних. Скалярні структури даних. Комбіновані структури даних. Рівномірні та неоднорідні структури даних. Масиви.

Алгоритми зсуву масиву. Одно- та багатовимірне оголошення та зберігання масиву. Програмування дисплея консолі. Алгоритми переміщення 2-D масиву. Особливості роботи з квадратними матрицями.

Тема 4: Алгоритми з масивами.

Алгоритми пошуку. Алгоритм лінійного пошуку та його програмування. Алгоритми сортування та їх використання. Оцінка алгоритмів сортування. Сортування шляхом прямої вставки. Алгоритми сортування шляхом прямого вибору. Алгоритм швидкого сортування. Сортування злиттям. Сортування з підрахунком. Апаратне сортування. Алгоритми з рядками. Функції з рядками. Структури. Множини. Юніони.

4. Навчальні матеріали

4.1. Базова література

1. Керниган Б., Рітчі Д. Мова програмування Сі. 229 с.
2. Кормен Т. Г., Лейзерсон Ч. Е., Рівест Р. Л., Стайн К. Вступ до алгоритмів. Київ: К. І. С., 2019. 1288 с.
3. Макконелл. Дж. Основи сучасних алгоритмів. 2004. 368 с.

4.2. Додаткова література

4. Кормен Т. Г. Алгоритми доступно. Київ: К.І.С, 2021. 206 с.

5. Ковалюк Т. В. Основи програмування. Київ: BHV. 2005. 384 с.

6. Непийвода Н. Н., Скопін І. Н.. Основи програмування. 2002. 919 с.

7. Вірт, Алгоритми та структури даних Н.. 358 с. ДМК. 2022. 272 с.

8. Sergiyenko A., Serhienko A., Romankevich V. "Genetic Programming of Pipelined Datapaths for FPGA," *2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO)*, Kyiv, Ukraine, 2020, pp. 802-806, doi: 10.1109/ELNANO50318.2020.9088773.

4.3. Інформаційні ресурси

На платформі дистанційного навчання «Сікорський»: https://classroom.google.com/c/NDIyNTg2NDY3NTY4?cjc=whbtqcx

Курс відеолекцій – на платформі дистанційного навчання Big Blue Button: https://bbb.comsys.kpi.ua/b/ana-fxo-y2d-ykw

Слайди лекцій та керівництва до лабораторних робіт https://kanyevsky.kpi.ua/en/for-students/

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Навчальний зміст дисципліни складається з лекцій та лабораторних робіт, які виконуються у лабораторному практикумі.

Лекції.

Тема 1: Основи мови С.

Лекція 1. Вступ.

Предмет і мета курсу «структур даних та алгоритмів». Сучасний рівень обчислювальної техніки. Навчальні матеріали. Програмне забезпечення для використання. Рейтингова система оцінок. Історія алгоритмів та розрахунків. Концепція алгоритму. Обчислювальний процес. Граф алгоритму. Архітектура комп’ютера та його історія. Ієрархія архітектури. Архітектура з точки зору програміста.

Лекція 2. Модель мови С.

Модель фон Неймана. Історія мови С. Стандарти мови С. Мова С для комп'ютера PDP-11. Модель мови С. Об'єкти мови С: коментарі, символи, ключові слова, ідентифікатори, літерали. Дані та їх види. Змінні. Локальні та глобальні об’єкти.

Лекція 3. Операції з мовою С

Операції та оператори. Арифметичні та логічні операції. Операції призначення. Пріоритети операцій. Перетворення типів у виразах.

Тема 2: Алгоритми та програми.

Лекція 4. Конструкції для проектування алгоритму

Блокові конструкції. Конструкції управління: конструкції розгалуження, повторення, перехід. Умовні, безумовні та та селективні конструкції. Однорівневі конструкції переходів.

Лекція 5. Подання алгоритмів та оператори.

Діаграми діяльності. Блок -схеми алгоритмів. Оператори умовного та безумовного переходу. Функції вводу-виводуі. Теорія схем програм. Основи структурного програмування.

Лекція 6. Основи оцінки алгоритмів.

Часова складність. Просторова складність. Стандартні моделі для оцінки часової складності. Розрахунок часової складності в програмах С. Оцінка порядку складності. Ітеративні алгоритми. Приклад алгоритму та розрахунок його складності.

Лекція 7. Основи оцінки алгоритму (продовження)

Спрощена оцінка складності. Типи алгоритмів: поліноміальна складність та NP-повні алгоритми. Вимірювання складності в програмі. Приклад алгоритму та розрахунок його складності.

Лекція 8. Оптимізація алгоритму.

Найпростіші алгоритмічні прийоми. Використання множника зі змінним знаком. Перевірка парних номерів. Використання прапорця знака. Розгортання циклів. Заміна множення на додавання. Метод динамічного програмування. Мінімізація умовних гілок.

Лекція 9. Підпрограми.

Функції та процедури. Контекст, інтерфейс та реалізація підпрограми. Виклик підпрограми. Реалізація підпрограми в архітектурі. Методи передачі параметрів. Функції на мові С. Структура програми С. Програмування функцій у С. Передача параметрів за значенням. Параметр, що передається за посиланням. Параметр типу const. Ключове слово static. Конвенція про виклик підпрограм у мові С.

Тема 3: Структури даних.

Лекція 10. Структури даних.

Структури даних. Концепція абстрактного типу даних. Функції над структурами даних. Класифікація структур даних. Статичні та динамічні структури даних. Скалярні структури даних. Комбіновані структури даних. Рівномірні та неоднорідні структури даних. Масиви. Одновимірний масив, його зберігання та доступ.

Лекція 11. Масиви та їх зміна.

Алгоритми зсуву масиву. Циклічний зсув масиву в одну позицію. Циклічний зсув масиву на k позицій. Обмін блоками. Циклічний зсув, що зберігає середню частину масиву. Циклічний зсув із переворотами.

Лекція 12. Двовимірні масиви

Оголошення та зберігання багатовимірного масиву. Ініціалізація двовимірного масиву. Програмування дисплея консолі. Бібліотека Windows та її функції для програмування прямого доступу до буфера консолі. Алгоритми обходу двовимірного масиву. Особливості роботи з квадратними матрицями.

Тема 4: Алгоритми з масивами.

Лекція 13. Алгоритми пошуку.

Задача пошуку. Алгоритм лінійного пошуку та його програмування. Класичний алгоритм лінійного пошуку. Лінійний пошук з бар’єром. Двійковий пошук та його дві варіації. Прямий пошук рядків. КМП-пошук. Алгоритм Рабіна-Карпа. Алгоритми пошуку з хеш-функцією.

Лекція 14. Алгоритми сортування.

Алгоритми сортування та їх використання. Оцінка алгоритмів сортування. Сортування прямою вставкою. Алгоритм з лінійним пошуком за вставленим елементом. Алгоритм з лінійним пошуком від елемента, який вставляється. Алгоритм з лінійним пошуком від елемента, що вставляється з бар'єром. Алгоритм з двійковим пошуком. Алгоритми сортування шляхом прямого вибору. Аналіз алгоритмів прямої вставки та прямого вибору.

Лекція 15. Алгоритми сортування ((закінчення)

Алгоритм прямого обміну. Алгоритм бульбашкового сортування. Алгоритм шейкерного сортування. Алгоритм Шела. Алгоритм швидкої сортування. Сортування злиттям. Сортування з підрахунком. Апаратне сортування. Мережа бітонного сортування. Мережа бульбашкового сортування. Мережа сортування парно-непарним злиттям.

Деталізація тем лекційних занять, конспект лекцій та відеоматеріали розміщено на платформі дистанційного навчання «Сікорській»
https://classroom.google.com/c/NDIyNTg2NDY3NTY4?cjc=whbtqcx

Лабораторні роботи

Мета лабораторних робіт – експериментальне дослідження та випробування алгоритмів та структур даних, що вивчаються, при програмуванні завдань лабораторних робіт.

Тематика лабораторних робіт:

Лабораторна робота 1.
Розгалужені алгоритми.
Лабораторна робота 2.
Циклічні алгоритми та динамічне програмування.
Лабораторна робота 3.
Алгоритми обходу двовимірних матриць.
Лабораторна робота 4.
Двійкові алгоритми пошуку.
Лабораторна робота 5.
Алгоритми сортування.

Загальні методичні рекомендації для дистанційного виконання:

1. **Зосередитись на програмуванні.** Основний навчальний чинник курсу — експериментальне випробування алгоритмів та структур даних, що вивчаються, при програмуванні завдань лабораторних робіт на власному комп'ютері студента. При цьому програмування можна виконувати у будь-яких сприятливих умовах у довільний час. При відсутності центрального електропостачання, програмування можна виконувати використовуючи заряд акумулятора.
2. **Використання матеріалів, які доступні в мережі Інтернет.** При неможливості бути присутнім на лекції кожну лекцію можна продивитись у запису, який зберігається на сервері кафедри, у довільний час, який є зручним для студента, навіть з повтором, якщо щось є незрозумілим з першого разу. Також в Інтернеті є багато довідкового матеріалу, що стосується курсу, який можна знайти за пошукачем чи у Вікіпедії або користуючись генеративним штучним інтелектом.
3. **Комунікація та підтримка.** Використовувати онлайн-інструменти для зворотного зв'язку, обговорення завдань та підтримки студентів у процесі виконання проєктів. Важливо мати регулярні консультації та можливість допомогти студентам у разі виникнення питань.

6. Самостійна робота здобувача вищої освіти

Види самостійної роботи для студентів денної форми навчання (76 годин):

• підготовка до лекцій шляхом вивчення попереднього лекційного матеріалу, а також літературних джерел, на яких ґрунтується матеріал попередніх лекцій (1 година х 15 лекцій = 15 годин);

• підготовка до лабораторної роботи через ознайомлення із завданням та методичними вказівками до лабораторних робіт, включаючи вивчення теоретичного матеріалу, необхідного для відповіді на контрольні запитання до лабораторних робіт, складання і програмування алгоритму, компіляцію та відлагодження програми, розгортання програмного середовища та завантаження вихідних кодів програмного забезпечення для виконання лабораторних робіт, робота з системою контролю версій Git, а також оформлення звіту. Кожна лабораторна робота вимагає для підготовки та виконання двох тижнів і витрат до 7 годин часу (5 х 7 годин = 35 годин);

• самостійне опрацювання теоретичного матеріалу (20 годин);

• підготовка до МКР (6 годин).

Теми на самостійне опрацювання

Тема 1: Основи мови С.

Основи роботи з IDE. Введення в систему керування версіями Git. (5 годин).

Тема 2: Алгоритми та програми.

Модель машини Тюрінга і основи оцінки алгоритму. Підпрограми з динамічним зв’язуванням. Взаємодія програми з операційною системою.Передача адрес через дескриптори. (5 годин).

Тема 3: Структури даних.

Файлові структури даних. Функції роботи з файлами. (5 годин).

Тема 4: Алгоритми з масивами.

Алгоритми з рядками. Рядки. Представлення рядків в комп'ютерах. Функції з рядками. Записи, структури. Множини. Юніти. (5 годин).

Матеріали для самостійного опрацювання, теми, питання для самоконтролю та контрольні запитання знаходяться на платформі дистанційного навчання «Сікорський» (Classroom). https://classroom.google.com/c/NDIyNTg2NDY3NTY4?cjc=whbtqcx

Політика та контроль

7. Політика навчальної дисципліни (освітнього компонента)

Система вимог до студентів:

• студент зобов’язаний відвідувати лекції та заняття комп'ютерного практикуму, активно працювати над засвоєнням викладеного на них матеріалу;

• на лекції лектор використовує власний презентаційний матеріал;

• лабораторні роботи захищаються у два етапи: перший етап - студенти виконують завдання, складають електронний звіт та надсилають викладачу; другий етап - захист лабораторних робіт у лабораторії. Контроль знань на лабораторних роботах здійснюється шляхом перевірки звіту про лабораторні роботи, а також питань з матеріалів робіт.

• модульна контрольна робота пишеться як тест з використанням усіх наявних матеріалів, але без використання комп’ютерних засобів.

Дедлайни. Для виконання лабораторних робіт встановлюються дедлайни. Дедлайни уточнюються і оприлюднюються викладачем на початку семестру.

Заохочувальні бали (бонуси) виставляються за: активну участь на лекціях та лабораторних заняттях, ведення конспекту, підготовку повідомлення з презентацією по одній із тем СРС дисципліни тощо. Кількість заохочуваних балів не більше 5.

Лабораторні роботи. Виконанню кожної лабораторної роботи передує реалізація індивідуального завдання і оформлення його у вигляді протоколу. Протокол виконання лабораторного завдання має бути вислано на пошту викладача не пізніше ніж за добу до лабораторного завдання. До протоколу додаються посилання на Git з кодом програм (за необхідності) і посилання на всі необхідні супровідні матеріали для захисту, зокрема на відеозапис виконання лабораторного завдання (за необхідності, в дистанційному режимі або за використання віддаленого устаткування, тощо). Студент, який відвідав заняття без підготовленого протоколу до захисту лабораторної роботи не допускається. Першим етапом студент захищає результати виконання індивідуального завдання до лабораторної роботи, на другому етапі – захищає теорію шляхом усного опитування або тестування. В дистанційному режимі більшість лабораторних робіт супроводжуються тестами для оцінки вивченого теоретичного та лабораторного матеріалу замість усного опитування. Бали отримані за виконання лабораторної роботи - за тест та за протокол складають оцінку за лабораторну роботу. Тестування проводиться на лабораторному занятті після перевірки результатів виконання індивідуальної роботи. Студент, який не виконав індивідуальне завдання до лабораторної роботи до тесту не допускається.

Виконання лабораторних робіт та МКР є обов'язковими для допуску до семестрового контролю (заліку). Загальною умовою допуску до семестрового контролю (заліку) є зарахування всіх лабораторних робіт, МКР та загальний рейтинг не менше 60 балів.

Оцінка, яку студент може отримати за виконання кожної лабораторної роботи наведені у розділі 8 силабуса.

Здобувачі, які виконали всі умови допуску до заліку та мають рейтингову оцінку менше 60 балів, а також здобувачі, які бажають підвищити свою рейтингову оцінку, на останньому за розкладом занятті складають залік у вигляді виконання залікового завдання та співбесіди.

З метою посилення зацікавленості здобувачів у якісному виконанні індивідуальних семестрових завдань, передбачених індивідуальним навчальним планом здобувача, рейтингова (семестрова) оцінка, у разі виконання залікової контрольної роботи, визначається, як оцінка за залікову контрольну роботу та сума балів за індивідуальні семестрові завдання (що відповідають балам за лабораторні роботи протягом семестру). Всі інші бали, які набрані протягом семестру (окрім балів за лабораторні роботи) анулюються.

Модульна контрольна робота є обов'язковою для допуску до семестрового контролю. Модульна контрольна робота (МКР) пишеться на лекційному занятті без застосування допоміжних засобів (мобільні телефони, планшети та ін.); результат пересилається до відповідної директорії Google -диску через Google форму.

У разі виконання модульної контрольної роботи рейтингова (семестрова) оцінка визначається як сума балів за модульну контрольну роботу, балів за індивідуальні семестрові завдання (лабораторні роботи).

До індивідуальних семестрових завдань вноситься індивідуальна робота студента що стосується виконання лабораторних робіт. Кількість балів за індивідуальну роботу за семестр дорівнює від 38 до 60 балів. Максимальна оцінка за МКР дорівнює 40 балів.

Правила відвідування занять. Відвідування занять є вільним, бали за присутність на лекціях та лабораторних заняттях не додаються, відповідно штрафні бали не передбачаються. Втім, вагома частина рейтингу формується через активну участь у заходах на лабораторних заняттях.

Дистанційний режим навчання. У разі запровадження обмежень на відвідування університету, пов'язаних з введенням карантину або режиму воєнного стану в державі, освітній процес здійснюється у дистанційному режимі відповідно до Положення про дистанційне навчання в КПІ ім. Ігоря Сікорського (https://osvita.kpi.ua/index.php/node/188), Регламенту організації освітнього процесу в дистанційному режимі (https://profkom.kpi.ua/reglament-organizatsiyi-osvitnogo-protsesu-v-distantsiynomu-rezhimi) та Регламенту проведення семестрового контролю в дистанційному режимі (https://osvita.kpi.ua/node/148). У режимі дистанційного навчання заняття відбуваються у вигляді онлайн-конференції на платформах BigBlueButton, Google Meet, Zoom.

Посилання на конференцію видається на початку семестру і розміщується в АС «Електронний кампус». З метою забезпечення якісної підготовки здобувачів, дистанційний курс дисципліни розміщено на Платформа дистанційного навчання «Сікорський» (https://www.sikorsky-distance.org). Результати оцінювання висвітлюють у АС «Електронний кампус» на особистій сторінці здобувача (https://ecampus.kpi.ua).

Правила поведінки на заняттях. На заняттях слід дотримуватись норм етичної поведінки визначених у Кодексі честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (https://kpi.ua/code). На території університету здобувачі мають поводити себе відповідно до Правил внутрішнього розпорядку (https://kpi.ua/admin-rule).

Визнання результатів навчання, набутих у неформальній/інформальній освіті. Положення про визнання результатів навчання, набутих у неформальній / інформальній освіті (https://osvita.kpi.ua/index.php/node/179) регламентує визнання результатів навчання, набутих у неформальній/інформальній освіті.

Неформальна освіта. У разі проходження здобувачем онлайн курсу чи іншого елементу неформальної освіти здобувачеві/здобувачці можуть бути зараховані окремі змістовні модулі або теми дисципліни. В такому разі здобувач звільняється від виконання відповідних лабораторних робіт, отримуючи за них максимальний бал відповідно до рейтингової системи оцінювання. У разі навчання на неформальній освіті за самостійного її обрання проходиться процедура валідації, що передбачає подання здобувачем заяви на ім'я декана, декларації підтверджувальних документів. Рішення про визнання чи не визнання приймається комісією у складі завідувача кафедри, викладача, гаранта освітньо-професійної програми.

Політика використання штучного інтелекту. Використання штучного інтелекту регламентується «Політикою використання штучного інтелекту для академічної діяльності в КПІ ім. Ігоря Сікорського» (https://osvita.kpi.ua/node/1225). Усі завдання, як під час виконання навчальних завдань з дисципліни, так і індивідуальні завдання, мають бути результатом власної оригінальної роботи здобувача. Використання ШІ для автоматичної генерації відповідей без подальшого їх аналізу та доопрацювання заборонено.

Порушення академічної доброчесності: використання плагіату. У разі виявлення фактів використання плагіату під час виконання навчальних завдань, студент втрачає право на подальше виконання поточних завдань у межах навчальної дисципліни до з'ясування обставин порушення. Студент зобов'язаний аргументовано пояснити причини використання плагіату, а також засвідчити готовність дотримуватись принципів академічної доброчесності в подальшому навчанні; повторно виконати завдання або, за потреби, виконати додаткове завдання відповідно до бачення викладача з урахуванням вимог щодо самостійності.

Зазначені дії регламентуються Кодексом честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (https://kpi.ua/code)

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

На першому занятті студенти знайомляться з політикою оцінювання, яка ґрунтується на Положенні про систему оцінювання результатів навчання https://document.kpi.ua/files/2020_1-273.pdf. Рейтинг студента на курсі складається з балів, які він/вона отримує за захищені лабораторні роботи (R1) та модульну контрольну роботу (R2).

Rs = R1+R2 = 100 балів

В результаті максимальний бал дорівнює:

5 лабораторних робіт х 12 балів = 60 балів

модульна контрольна робота = 40 балів.

Згідно з університетськими правилами щодо моніторингу успішності студента (https://kpi.ua/document_control), існує два тижні оцінювання (атестація), зазвичай протягом 7/8 та 14/15 тижнів семестру, коли оцінюється прогрес у навчанні студентів та відповідність до критеріїв політики оцінювання курсу.

Умовою першої атестації є отримання не менше 10 балів (на момент атестації). Умова другої атестації - набрати не менше 30 балів (на момент атестації).

Критерії оцінювання лабораторних робіт наступні:

Протокол: Відповідність протоколу лабораторній роботі та встановленим вимогам до оформлення, коректність розміщення протоколу в Classroom/Git, повнота та логічність викладення матеріалу:

0 балів – протокол відсутній або не відповідає лабораторній роботі та встановленим вимогам, зокрема містить скриншоти та інші ілюстративні матеріали, які не дозволяють однозначно ідентифікувати середовище та користувача виконання роботи.

1 бал – протокол частково відповідає лабораторній роботі та вимогам до оформлення; матеріал подано неповно або з порушенням логіки викладення, результати виконання неповні, відсутні висновки.

2 бали – протокол повністю відповідає лабораторній роботі та встановленим вимогам до оформлення; коректно розміщений у Classroom/Git; матеріал викладено повно й логічно, наявні змістовні висновки; надані ілюстративні матеріали (скриншоти, фрагменти консолі тощо) є автентичними, належать автору роботи та містять ідентифікаційні ознаки користувача середовища виконання.

Виконання завдання: розроблена програма виконується правильно, відповідає вимогам завдання (5 балів), те саме, але не всі вимоги виконуються (4 бали), програма виконується з деякими помилками (2-3 бали), програма не виконується (0 балів);

Опитування/тестування за тематикою лабораторної роботи для зарахування теоретичної частини роботи, відповіді на додаткові теоретичні запитання викладача: 0 – 5 балів.

Критерії оцінювання усної відповіді:

Знання теоретичного матеріалу	Студент правильно відповідає на запитання за тематикою лабораторної роботи, демонструє розуміння ключових понять.	1–2 бали
Повнота та логічність відповіді	Відповіді послідовні, зрозумілі, демонструють усвідомлення зв’язку теорії та практики.	1 бал
Відповіді на додаткові запитання викладача	Студент коректно та аргументовано відповідає на додаткові теоретичні запитання, пояснює нюанси виконаного завдання.	1–2 бали

• Модульна контрольна робота оцінюється 40 балами. Вона проводиться як автоматизоване тестування на платформі Google Workspace for Education. Контрольна робота складається з 16 тестових питань, а також практичного завдання (скласти програму) зі списку, наведеного в додатку до робочої програми.

За кожну правильну відповідь на тестове питання нараховується 2 бали. Відповідь на практичне завдання оцінюється 8 балами за такими критеріями:

- "відмінно" - правильний текст програми з коментарями - 8 - 7 балів;

- "добре" - текст програми, загалом правильний, але не використані прийоми оптимізації - 6-5 балів;

- "задовільно" - є деякі принципові помилки в тексті програми - 4 - 3 бали;

- "незадовільно" - незадовільна відповідь - 0 балів.

Студенти, які нарешті наберуть необхідну кількість балів (≥60), можуть:

- отримати остаточну залікову оцінку відповідно до рейтингового балу;

- виконати залікову контрольну роботу, щоб підвищити оцінку.

Студенти можуть отримати до 6 заохочувальних балів за виконання творчих робіт з кредитного модуля (складання тез, участь у конкурсах, у дослідженнях тощо).

Студенти, чий остаточний бал за успішність Rs нижче 60 балів, але більше за 30, повинні виконати залікову контрольну роботу. Студенти, чий остаточний бал за успішність Rs вище або дорівнює 60 балів, можуть виконати залікову контрольну роботу, щоб підвищити оцінку. В обох випадках застосовується вимога - попередній рейтинг студента складає: 0.7R1. Студенти, чий бал нижче 30, не допускаються до заліку.

Максимальна оцінка R3 за залікову контрольну роботу складає 60 балів. Контрольна робота проводиться у вигляді автоматизованого тестування на платформі Google Workspace for Education. Вона складається з 25 тестових питань, а також практичного завдання (скласти програму). За кожну правильну відповідь на тестове питання нараховується 2 бали. Відповідь на практичне завдання оцінюється 10 балами за такими критеріями:

- "відмінно" - правильний текст програми з коментарями - 10 - 9 балів;

- "добре" - текст програми, загалом, правильний, але не використані прийоми оптимізації - 8-6 балів;

- "задовільно" - є деякі принципові помилки в тексті програми - 5 - 4 балів;

- "незадовільно" - незадовільна відповідь - 0 балів.

Рейтингова оцінка разом з оцінкою за залікову контрольну роботу дорівнює

$Rs = 0.7R1 + R3$.

Таблиця. Остаточна оцінка RS приймається за системою оцінювання університету

Кількість балів	*Оцінка*
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

9. Додаткова інформація про курс

Питання в модульній контрольній роботі виглядають як наступні.

І 1. Обчислювальний алгоритм – це а) послідовність дій, виконання якої приводить до очікуваного результату, б) обчислювальний процес, що заданий на чітко описаній моделі обчислювача, в) граф або програма, обхід частин яких відповідає деякому обчислювальному процесу.

2. В ЕОМ використовуються 6-бітові літери. Алфавіт цієї ЕОМ вміщує: а) 32 цифри і знаки, б) 55 цифр, букв і знаків, в) 65 цифр, букв і знаків.

3. Що означає у мові Сі запис: \0 — а) ділення на нуль без переривання, б), символ NULL в) ознака кінця рядка?

4. Змінна типу int займає в машині об'єм – а) 16 біт, б) 32 біт, в) 16 або 32 біт .

5. Вираз: a&&b/c & d;еквівалентний виразу — а) (a&&(b/c)) & d;, б) a&&((b/c) & d);,в) (a&&b)/(c&d);.

6. Результатом операції 12 || 6 є – а) 6 , б) 1 , в) 14 .

7. Результатом операції switch (3 >>1){case 0: a = 1; case 1: a= 2; break; default: a=3;} є:а) 1, б) 2, в)3.

8. Блок — це: а) виділена частина програми, змінні якої мають локальну дію, б) виділена частина програми, яка має інтерфейс для вхідних і вихідних змінних, в) частина структури комп'ютера.

9. цикл з передумовою – це цикл, який а) перестає виконуватись за умовою, б) виконується за умовою, в) це цикл, який починає виконуватись за умовою .

10. Щоб цикл з лічильником не зациклився, слід забезпечити – а) умови обов'язкового виходу з циклу, б) щоб справджувалась умова оператора for хоча б в одній з ітерацій, в) щоб змінна лічильника була цілого типу.

11. Оператор continue призначений для – а) продовження роботи програмного модуля після переривання, б) виходу з циклу, в) переходу на наступну ітерацію циклу.

12. Оператор for (i = 1; ;i++){...} — виконує цикл, – а) оператор є неправильним, б) якщо у його блоці є оператор виходу з циклу, в) якщо у його блоці є оператор виходу з циклу, інакше — нескінченний цикл.

13. При виконанні фрагменту: int A[3] = {1,2,3}; printf("%d",A[3]); буде виводитись – а) 3 , б) 0 , в) невідоме число.

14. Дані статичної структури – це дані, – а) які складаються з констант, б) взаєморозташування та взаємозв'язки яких є постійними, в) які мають незмінне розташування під час виконання програми.

15. У програмі на Сі масив задано як int a[2][3] = { 1, 2, 3, 4, 5, 6 }; Чому дорівнює елемент a[2][3] а) NULL, б) невідоме число, в) 6 ?.

16. Сортування з підрахунком ґрунтується на тому, що підраховується – а) кількість елементів у масиві однакової величини, б) дистанція, на яку слід перемістити черговий елемент на черговому кроці впорядкування масиву, в) довжина частини масиву, яка потребує сортування.

II Розробити алгоритм і програму, що вирішує задачу. Є число з плаваючою комою X. Розрахувати масив X[n], у якому X[0] = X; $X[i] = \sum_{k=1}^{i}(-X)^k \ (k+1)$; Число операцій має бути мінімізоване.

Робочу програму навчальної дисципліни (силабус):

Складено д.т.н., проф., А. М. Сергієнко

Ухвалено кафедрою обчислювальної техніки (протокол №18 від 24.06.2026)

Погоджено Методичною комісією ФІОТ (протокол №12 від 26.06.2026)