

**ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД
“УЖГОРОДСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ”
ІНЖЕНЕРНО-ТЕХНІЧНИЙ ФАКУЛЬТЕТ
КАФЕДРА КОМП’ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ**

ЗАТВЕРДЖУЮ

**Декан інженерно-технічного
факультету**

доц. Йолана ГОЛИК.

“ 20 ” червня 2025 р.

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
СТРУКТУРИ ДАНИХ ТА АЛГОРИТМИ**


**Рівень вищої освіти – перший (бакалаврський)
Галузь знань – F – Інформаційні технології
Спеціальність – F7 – Комп’ютерна інженерія
Освітня програма – Комп’ютерні системи та мережі
Статус дисципліни – обов’язкова
Мова навчання – українська**

Ужгород – 2025


Робоча програма навчальної дисципліни «Структури даних та алгоритми» для здобувачів вищої освіти галузі знань F «Інформаційні технології» спеціальності F7 «Комп'ютерна інженерія» освітньої програми «Комп'ютерні системи та мережі» – 22 с.

Розробник: Самусь Є.І., старший викладач кафедри комп'ютерних систем та мереж

Робочу програму розглянуто та затверджено на засіданні кафедри комп'ютерних систем та мереж протокол 13 від «25» червня 2025 р.

Завідувач кафедри  доц. Петро ГОРВАТ
(підпис)

Схвалено науково-методичною комісією інженерно-технічного факультету протокол 6 від «27» червня 2025 р.

Голова науково-методичної комісії  доц. Володимир ЦИГИКА
(підпис)

1. ОПИС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Найменування показників	Розподіл годин за навчальним планом	
	денна форма навчання	
Кількість кредитів ЄКТС – 7	Рік підготовки:	
Загальна кількість годин – 210	1-й	
Кількість модулів – 4	Семестр:	
	1-й	2-й
Тижневих годин: аудиторних – 3,3/2,9 год самостійної роботи студента – 3,3/3,1 год	Лекції:	
	30 год.	22 год.
	Практичні заняття:	
	14 год.	
	Лабораторні заняття:	
	16 год.	22 год.
Вид підсумкового контролю: залік, екзамен	Самостійна робота:	
	60 год.	46 год.
Форма підсумкового контролю: усна	Індивідуальні завдання: –	

2. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Мета дисципліни - ознайомлення студентів з основними, класичними структурами даних та алгоритмами, які використовуються при програмуванні, навчання студентів навичкам проектування складних програм з використанням сучасних технологій структурного програмування.

Завдання дисципліни - розширення світогляду студентів відносно можливостей програмування, формування професійного відношення до створення програмного продукту, розуміння того факту, що тільки ретельно продуманий вибір необхідної структури даних та обчислювальних алгоритмів для конкретної задачі може бути гарантією написання програми, що відповідає сучасним вимогам.

Програма містить перелік тем, питань, які розглядаються на лекціях, практичних та лабораторних заняттях. Програмою передбачена самостійна робота студентів та контроль за нею. Приводиться список основної та допоміжної літератури, яка рекомендується для вивчення цієї дисципліни.

Відповідно до освітньої програми вивчення дисципліни сприяє формуванню у здобувачів вищої освіти таких компетентностей:

- інтегральна (здатність розв'язувати складні задачі і проблеми у певній галузі професійної діяльності або у процесі навчання, що передбачає проведення досліджень та/або здійснення інновацій та характеризується невизначеністю умов і вимог);

- загальні (ЗК1- здатність до абстрактного мислення, аналізу і синтезу, ЗК2 - здатність вчитися і оволодівати сучасними знаннями, ЗК3-здатність застосовувати знання у практичних ситуаціях, ЗК7- вміння виявляти, ставити та вирішувати проблеми, ЗК8- здатність працювати в команді.);

- фахові (ФК2-здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення, ФК3- здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж, ФК11-здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів, ФК13-здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій, ФК15-здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення).

3. ПЕРЕДУМОВИ ДЛЯ ВИВЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Передумовами вивчення навчальної дисципліни “Структури даних та алгоритми” є опанування курсів “Програмування”, “Організація та функціонування комп'ютерів”.

4. ОЧІКУВАНІ РЕЗУЛЬТАТИ НАВЧАННЯ

Відповідно до освітньої програми вивчення навчальної дисципліни «Структури даних та алгоритми» повинно забезпечити досягнення здобувачами вищої освіти таких програмних результатів навчання (ПРН):

Програмні результати навчання	Шифр ПРН
Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж.	ПРН1
Мати навички проведення експериментів, збирання даних та моделювання в комп'ютерних системах.	ПРН2
Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, спеціальності обладнання.	ПРН10
Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії .	ПРН11
Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення.	ПРН16

Очікувані результати навчання, які повинні бути досягнуті здобувачами освіти після опанування навчальної дисципліни „ Структури даних та алгоритми”:

Очікувані результати навчання з дисципліни	Шифр ПРН
Знати і розуміти основні поняття інформатики, математики, основні структури даних, дані з динамічною структурою, алгоритми маніпулювання даними різної структури, методи структурного та модульного програмування і проектування, різні способи запису алгоритмів. Вміння шукати необхідну інформації згідно отриманого завдання на лабораторних та практичних заняттях, згідно додаткових завдань, аналізувати та оцінювати її.	ПРН1 ПРН2 ПРН11
Розробляти складні алгоритми на основі вивчених основних алгоритмів та структур даних з використанням структурних методів розробки програм, вміння розв'язувати спектр проблем при вирішенні практичних задач від проблеми формалізації задачі до проблем, які постають у час виконання закінченої програми, застосовувати вибраний або розроблений алгоритм до конкретних вихідних даних задачі, яка вирішується.	ПРН10
Презентувати, обговорювати та захищати власний вибір та засоби реалізації проєктів виконаних в усній чи електронній формах в межах лабораторних робіт та індивідуальних завдань, підтверджувати правильність рішень відповідними тестовими прикладами.	ПРН16

5. ЗАСОБИ ДІАГНОСТИКИ ТА КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ НАВЧАННЯ

Робоча програма з дисципліни «Структури даних та алгоритми» містить 4 модулі. Використовуються методи усного контролю та письмового контролю. Поточний контроль передбачає: опитування студентів під час захисту лабораторних робіт та опитування на лекціях і практичних заняттях; контрольні роботи, індивідуальні, самостійні та тестові завдання. Підсумковий контроль передбачає залік у першому семестрі та екзамен у другому семестрі.

Для контролю знань розроблено: перелік теоретичних питань та типових завдань, (наведено в додатку робочої програми); завдання для самостійної роботи, зі змістом яких студенти ознайомлюються на початку семестру.

Оцінка ECTS, яку студент отримує після вивчення кредитного модуля дисципліни, визначається відповідно до рейтингу студента. Рейтинг студента з кредитного модуля складається з балів, що він отримує протягом семестру за такі види робіт:

1. Модульна контрольна робота (МКР) тривалістю по 2 акад. години. Максимальна кількість балів за МКР – 40 балів.

2. Виконання лабораторних робіт – максимальна кількість 50 балів.

3. Бали із індивідуальної та самостійної роботи студентів нараховуються за: підготовку рефератів, модернізацію завдань, за творчий підхід до виконання завдань, за виконання завдань підвищеної складності, за удосконалення дидактичних матеріалів з дисципліни: 1-10 балів за кожен модуль.

Сума вагових балів контрольних заходів протягом семестру: 100 балів.

Необхідною умовою допуску до іспиту є відсутність заборгованостей з лабораторних робіт та зарахування контрольних робіт. У кінці вивчення дисципліни виводиться рейтинговий бал, який визначається як середньоарифметичне балів отриманих за кожний модуль.

Оцінювання окремих видів навчальної роботи з дисципліни

Вид діяльності здобувача вищої освіти	Модуль 1	
	Кількість	Максимальна кількість балів (сумарна)
Лабораторні заняття (виконання та захист)	3	40
Самостійна робота	1	10
Модульна контрольна робота	1	50
Разом	5	100
Вид діяльності здобувача вищої освіти	Модуль 2	
	Кількість	Максимальна кількість балів (сумарна)
Лабораторні заняття (виконання та захист)	2	40
Самостійна робота	1	10
Модульна контрольна робота	1	50
Разом	4	100
Вид діяльності здобувача вищої освіти	Модуль 3	
	Кількість	Максимальна кількість балів (сумарна)
Лабораторні заняття (виконання та захист)	3	40
Самостійна робота	1	10
Модульна контрольна робота	1	50
Разом	5	100
Вид діяльності здобувача вищої освіти	Модуль 4	
	Кількість	Максимальна кількість балів (сумарна)
Лабораторні заняття (виконання та захист)	2	40
Самостійна робота	1	10
Модульна контрольна робота	1	50

Разом	4	100
-------	---	-----

Критерії оцінювання модульної контрольної роботи

Матеріал кожного модуля, який здобувачі вищої освіти повинні засвоїти протягом семестру, виноситься на одну з двох модульних контрольних робіт, що проводяться в семестрі згідно з розкладом затвердженим деканатом.

Модульна контрольна робота (при застосуванні комп'ютерів) може виконуватись в два етапи: перший – теоретичний матеріал (письмово); другий – практичні або тестові завдання (із застосуванням комп'ютерів). Також в залежності від форми навчання очної чи дистанційної, МКР може проводитись у двох режимах:

- письмова, яка містить шість завдань. Перші три завдання включають теоретичний і практичний матеріал, наступні 3 завдання – це тести чи практичні завдання.

- тестова, що містить 20-40 тестів із різними типами запитань, в основному з вибором однієї правильної відповіді.

Сумарна максимальна кількість балів, що виставляється здобувачу вищої освіти за виконання всіх завдань однієї контрольної роботи залежить від складності матеріалу, який виноситься на модульний контроль.

За виконання лабораторних робіт та завдань самостійної роботи, що стосуються даних модулів здобувачу вищої освіти нараховується різна кількість балів, в залежності від складності матеріалу.

Лабораторна робота зараховується, якщо за кожне із завдань лабораторної роботи здобувач вищої освіти досягнув мінімального порогового рівня, визначеного за кожним запланованим результатом навчання для навчальної дисципліни. Критерієм успішного проходження здобувачем освіти поточного оцінювання (включно із захистом лабораторної роботи та презентацією результатів) є досягнення здобувачем освіти не менше 50% балів від загальної кількості запланованої за конкретною темою. Конкретна максимальна кількість балів подається у таблицях розподілу балів, які отримують здобувачі за модуль та за окремі види навчальної роботи.

Невиконані та незахищені лабораторні роботи, а також неявка на модульну контрольну роботу оцінюються в 0 балів незалежно від причини невиконання (неявки). Сумарна оцінка (від 0 до 100 балів) виставляється у відомість модульного контролю. Модуль зараховується, якщо сумарний бал складає не менше 60 балів, і зараховано всі лабораторні роботи, які є складовими даного модуля. Здобувач вищої освіти, який не з'явився на модульну контрольну роботу, або ж його модульна оцінка складає від 0 до 34 балів, зобов'язаний скласти (перескласти) модуль до початку підсумкового контролю у строки, визначені викладачем дисципліни та погоджені деканатом факультету.

Критерії оцінювання підсумкового семестрового контролю

За результатами модульних контролів визначається підсумкова модульна оцінка. Залікова та екзаменаційна оцінка визначається в залежності від рейтингового балу, або балів за залік та екзамен.

До складання екзамену допускаються здобувачі вищої освіти, у яких підсумкова модульна оцінка за семестр становить не менше 35 балів і, яким зараховано всі лабораторні роботи за цей семестр.

Здобувач вищої освіти, підсумкова модульна оцінка якого складає від 0 до 34 балів, зобов'язаний покращити її до початку підсумкового семестрового контролю. В протилежному випадку, здобувач не допускається до екзамену і у нього виникає академічна заборгованість.

Екзамен з навчальної дисципліни здобувач вищої освіти може не складати, якщо він успішно пройшов усі модульні контролі та його влаштовує підсумкова модульна оцінка. Здобувачі вищої освіти, підсумкова модульна оцінка яких становить від 35 до 59, екзамен складають обов'язково. Здобувач освіти може підвищити на екзамені рейтинговий бал, при цьому, за результатами складання екзамену оцінка не може бути менша за підсумкову модульну оцінку, яку він отримав за результатами модульних контролів у відповідному семестрі.

На залік/екзамен виносяться теоретичні та практичні завдання в обсязі навчального матеріалу за семестр. Оцінювання результатів навчання на заліку здійснюється за 100-бальною шкалою. Оцінка за екзамен вноситься у відомість обліку успішності.

За результатами виконання студентом навчальної програми впродовж семестру рекомендується виставляти заліки та екзамени без додаткового опитування за такою шкалою:

Сумарні бали	Оцінка ECTS	Екзамен (диф.залік)	Залік
90 – 100	A	Відмінно	Зараховано
82 – 89	B	Добре	
74 – 81	C		
64 – 73	D	Задовільно	
60 – 63	E		
35 – 59	FX	Незадовільно з можливістю повторного складання	Незараховано з можливістю повторного складання
1 – 34	F	Незадовільно з обов'язковим повторним вивченням дисципліни	Незараховано з обов'язковим повторним вивченням дисципліни

Критерій оцінювання підсумкового контролю з дисципліни

— «відмінно» (90-100 балів, A) заслуговує здобувач, який виявив всебічне і глибоке знання програмового матеріалу, вміння вільно виконувати завдання, передбачені програмою, засвоїв основну і ознайомився з додатковою літературою, розуміє взаємозв'язок головних понять дисципліни та їх значення для майбутньої професії;

— «добре» (82-89 балів, B) заслуговує здобувач, який виявив повне знання програмного матеріалу, успішно виконує передбачені програмою завдання,

засвоїв основну літературу рекомендовану програмою, виявив систематичний характер знань з дисциплін і здатний до самостійного доповнення, але під час відповіді допустив деякі неточності;

— **«добре» (74-81 бал, C)** заслуговує здобувач, що виявив не цілком повне знання програмного матеріалу, не завжди успішно виконує передбачені програмою завдання, частково засвоїв основну літературу, рекомендовану програмою, виявив не систематичний характер знань з дисциплін і не завжди здатний до їх самостійного доповнення і під час відповіді допускає деякі неточності;

— **«задовільно» (64-73 бали, D)** заслуговує здобувач, що виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та майбутньої роботи за професією, вміє виконувати завдання, передбачені програмою, знайомий з основною рекомендованою літературою. Як правило, дана оцінка виставляється студентам, що допустили помилки у відповіді на заліку чи екзамені та при виконанні залікових або екзаменаційних завдань, але які володіють необхідними знаннями для їх усунення за допомогою викладача;

— **«задовільно» (60-63 балів, E)** заслуговує здобувач, що виявив часткове знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та майбутньої роботи за професією, не завжди вміє виконувати завдання, передбачені програмою, знайомий лише частково з основною рекомендованою літературою. Як правило, дана оцінка виставляється студентам, що допустили грубі помилки у відповіді на заліку чи екзамені та при виконанні залікових або екзаменаційних завдань, але які частково володіють необхідними знаннями для їх усунення за допомогою викладача.

— **«незадовільно» (35-59 балів, FX)** виставляється здобувачу, який виявив суттєві прогалини в знаннях основного програмного матеріалу, допустив принципові помилки у виконанні передбачених програмою завдань.

— **«незадовільно» (0-34 балів, F)** виставляється здобувачу коли протягом семестру він допустив грубі помилки у виконанні передбачених програмою завдань.

6. ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

6.1 Зміст навчальної дисципліни

Модуль1

Тема 1. Зміст базових понять: Предмет курсу. Історичні відомості про розвиток алгоритміки. Основні визначення. Властивості алгоритмів. Способи запису алгоритмів. Основні керуючі конструкції алгоритмів. Блок-схеми та діючі стандарти. Поняття про ефективність алгоритмів

Тема 2. Структурне програмування. Базові алгоритмічні структури. Додаткові алгоритмічні структури. Представлення додаткових структур через базові. Перетворення неструктурних алгоритмів.

Тема 3. Типи алгоритмів. Лінійні, розгалужені і циклічні алгоритми. Обчислення сум, добутків, складних функцій. Обчислення з заданою точністю. Оцінка точності результату.

Тема 4. Поняття структури даних. Представлення даних в комп'ютерних системах. Класифікація структур. Операції над структурами. Структурність даних і технологія програмування.

Тема 5. Числові типи даних та операції над ними. Прямий, обернений та доповняльний код. Виконання операцій у доповняльному коді. Цілочислові беззнакові типи. Цілочислові знакові типи та їх представлення.

Тема 6. Нормалізація і представлення дійсних чисел. Особливості комп'ютерної арифметики. Типи Float, Double та їх представлення в пам'яті. Граничні та спеціальні значення у дійсних типах. Двійково-десяткові типи.

Тема 7. Інші прості структури даних. Бітові типи. Логічний тип. Поняття про логічні операції. Символьний тип. Представлення ASCII та Unicode. Перелічувальний та інтервальні типи.

Тема 8. Адресація пам'яті. Одержання фізичної адреси в реальному режимі. Поняття покажчика (вказівника). Операції над покажчиками. Динамічна пам'ять та динамічні змінні.

Модуль 2

Тема 1. Множини. Числові множини. Символьні множини. Операції над множинами.

Тема 2. Вектори та масиви. Вектори. Логічна та фізична структура вектора. Масиви. Представлення масивів у пам'яті. Дескриптор масиву. Операції над масивами. Спеціальні масиви: симетричні, розріджені.

Тема 3. Основи аналізу ефективності алгоритмів. Час виконання програм, його вимірювання. Складність алгоритмів. Асимптотичні оцінки складності.

Тема 4. Методи пошуку у статичних структурах. Постановка задач пошуку та сортування на прикладі вектора. Послідовний або лінійний пошук. Бінарний пошук. Вдосконалені варіанти методів пошуку

Тема 5. Алгоритми сортування в основній пам'яті. Сортування векторів методами прямої вставки, прямого вибору та прямого обміну. Оцінки складності прямих методів. Порівняння прямих методів.

Тема 6. Вдосконалені методи сортування. Сортування Шелла. Сортування з використанням дерева. Піраміда та пірамідальне сортування. Швидке сортування. Сортування злиттям. Порівняння методів сортування.

Модуль 3

Змістовий модуль 6. Напівстатичні структури даних.

Тема 1. Стеки. Логічна структура. Моделювання стека за допомогою вектора. Програмна реалізація операцій над стеком. Стеки в обчислювальних системах.

Тема 2. Черги FIFO. Логічна структура та машинне представлення. Черги з пріоритетами. Черги в обчислювальних системах: застосування та проблеми. Деки, їх структура, представлення та застосування

Тема 3. Рядки. Логічна структура. Операції над рядками. Представлення рядків за допомогою векторів.

Тема 4. Пошук у тексті. Задача пошуку у тексті. Прямий пошук рядка. Алгоритм Кнута-Морріса-Пратта. Алгоритм Боуера-Мура. Аналіз та порівняння алгоритмів.

Тема 5. Лінійні списки. Класифікація та опис спискових структур. Лінійні однозв'язні списки. Представлення в пам'яті. Обробка, упорядкування та перебудова списків. Застосування списків для організації рядків, стеків та черг. Списки і таблиці. Спискові алгоритми відкритого хешування. Представлення списків векторами покажчиків. Статичні вектори покажчиків. Операції із списками, представленими статичними векторами покажчиків.

Модуль 4

Тема 1. Бінарні дерева. Поняття бінарного дерева. Впорядковані бінарні дерева. Побудова дерева. Обхід дерева. Пошук у дереві. Сортування за допомогою дерева. Способи представлення бінарних дерев: використання масивів, нелінійних розгалужених списків.

Тема 2. Методи пошуку в основній пам'яті. Застосування бінарних дерев до пошуку. Ідеально збалансовані бінарні дерева. AVL-дерева. Вставка та вилучення вершин у AVL-деревах. Дерева оптимального пошуку. Дерева цифрового пошуку.

Тема 3. Методи пошуку в масовій пам'яті. Сильно розгалужені дерева (дерева). Основні терміни. Представлення дерев. Представлення довільного дерева через бінарне. Застосування дерев. Класичні B(B)-дерева. Додавання та вилучення ключів. Вдосконалення B-дерев.

Тема 4. Графи. Різновиди графів. Основні поняття. Представлення графів. Способи обходу графа. Алгоритми побудови кістякового дерева. Алгоритми обчислення найкоротших відстаней.

Тема 5. Файлові структури даних. Фізичні особливості дискової пам'яті. Логічна організація даних на диску. Основні файлові системи. Файлові системи типу FAT. Файлові системи NTFS. Порівняння файлових систем. Типи файлів. Текстові та двійкові файли.

Тема 6. Сортування файлів. Послідовні файли, проблема сортування, методи зовнішнього сортування. Файли, які організовані розділами. Файли прямого доступу. Індeksi. Блочне хешування.

6.2 Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	Денна форма					
	у тому числі					
	Лекції	Практичні	Лабораторні	Індивідуальна робота	Самостійна робота	
1	2	3	4	5	6	7
Модуль 1						
Тема 1. Зміст базових понять	2	2				
Тема 2. Структурне програмування.	7	2	1			4
Тема 3. Типи алгоритмів	8	2	2			4
Тема 4. Поняття структури даних	2	2				
Тема 5. Числові типи даних та операції над ними	9	2	1	2		4
Тема 6. Нормалізація і представлення дійсних чисел	9	2	1	2		4
Тема 7. Інші прості структури даних	9	2	1	2		4
Тема 8. Адресація пам'яті	7	2	1			4
Разом за модулем 1	53	16	7	6	0	24
Модуль 2						
Тема 1. Множини	9	2	1			6
Тема 2. Вектори та масиви	12	2	2	2		6
Тема 3. Основи аналізу ефективності алгоритмів	11	2	1	2		6
Тема 4. Методи пошуку у статичних структурах	11	2	1	2		6

Тема 5. Алгоритми сортування в основній пам'яті	<i>13</i>	<i>3</i>	<i>2</i>	<i>2</i>		<i>6</i>
Тема 6. Вдосконалені методи сортування	<i>11</i>	<i>3</i>		<i>2</i>		<i>6</i>
Разом за модулем 2	<i>67</i>	<i>14</i>	<i>7</i>	<i>10</i>	<i>0</i>	<i>36</i>
Усього годин за перший семестр	<i>120</i>	<i>30</i>	<i>14</i>	<i>16</i>	<i>0</i>	<i>60</i>
Модуль 3						
Тема 1. Стеки	<i>6</i>	<i>2</i>				<i>4</i>
Тема 2. Черги FIFO	<i>8</i>	<i>2</i>		<i>2</i>		<i>4</i>
Тема 3. Рядки	<i>7</i>	<i>1</i>		<i>2</i>		<i>4</i>
Тема 4. Пошук у тексті	<i>8</i>	<i>2</i>		<i>2</i>		<i>4</i>
Тема 5. Лінійні списки	<i>11</i>	<i>2</i>		<i>3</i>		<i>6</i>
Разом за модулем 3	<i>40</i>	<i>9</i>		<i>9</i>		<i>22</i>

Модуль 4						
Тема 1. Бінарні дерева	10	3		3		4
Тема 2. Методи пошуку в основній пам'яті	10	2		2		6
Тема 3. Методи пошуку в масовій пам'яті	8	2		2		4
Тема 4. Графи	14	4		4		6
Тема 5. Файлові структури даних	8	2		2		4
Разом за модулем 4	50	13	0	13	0	24
Усього годин за другий семестр	90	22	0	22	0	46

6.3 Теми практичних занять

№ з/п	Назва теми	Кількість годин
1	Розробка простих алгоритмів з використанням структурного програмування	2
2	Прості структури даних (числові, дійсні, логічні та символічні типи)	2
3	Проведення оцінки ефективності та складності алгоритму	2
4	Реалізація методів пошуку у статичних структурах	2
5	Прості та вдосконалені алгоритми сортування	3
8	Робота з рядками. Реалізація пошуку в тексті.	3
	Разом	14

6.4. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
Модуль 1		
1	Блок-схеми алгоритмів. Розробка блок-схем типових алгоритмів.	2
2	Представлення цілочислових та дійсних типів даних	2
3	Використання структур, як представлення даних	4
Модуль 2		
5	Методи пошуку у статичних структурах	4
6	Порівняння методів сортування	4
Модуль 3		
7	Робота з напівстатичними структурами: стеки, черги	4
8	Бінарні дерева, побудова, пошук у дереві	4

9	Різновиди дерев та операції над ними	4
Модуль 4		
10	Представлення графів, обходи графів	4
11	Алгоритми на графах	6
	Разом	38

6.5 Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Структурне програмування	6
2	Розгалужені та циклічні алгоритми	8
3	Робота з різними типами даних	10
4	Адресація пам'яті	6
5	Робота з масивами	6
6	Аналіз ефективності алгоритмів	5
8	Алгоритми сортування	11
9	Хешування	6
10	Використання стеків та черг	8
11	Алгоритми пошуку у тексті	7
12	Робота з лінійними списками	7
13	Методи пошуку в основній та масовій пам'яті	10
14	Робота з графами	9
15	Окремі алгоритми на графах	7
	Разом	106

7. ІНСТРУМЕНТИ, ОБЛАДНАННЯ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ВИКОРИСТАННЯ ЯКИХ ПЕРЕДБАЧАЄ НАВЧАЛЬНА ДИСЦИПЛІНА

Навчання проводиться з використанням сайту електронного навчання <https://moodle.uzhnu.edu.ua/>. Середовище програмування Visual Studio Premium, Visual Studio Community, онлайн-компілятори https://www.onlinegdb.com/online_c++_compiler, <https://cpp.sh/>, https://www.w3schools.com/cpp/trycpp.asp?filename=demo_helloworld.

8. Рекомендована література

1. Структури даних та алгоритми: підручник. У 2-х ч. Ч. 1. [Електронний ресурс] / О. І. Марченко, О. О. Марченко; КПІ ім. Ігоря Сікорського. – Київ : Просвіта, 2024. – 268 с.
URL: <https://ela.kpi.ua/items/e0777bf4-835c-4bd1-b95b-3c787f1ed449>
2. Кренивч А.П. Алгоритми і структури даних. Підручник. – К.: ВПЦ "Київський Університет", 2021. – 200 с.
3. Томас Г. Кормен, Чарлз Е. Лейзерсон, Роналд Л. Рівест, Кліфорд Стайн. Вступ до алгоритмів. — К. : К. І. С., 2019. — 1288 с.
4. Алгоритми і структури даних: навч. посіб. / Т. О. Коротеєва ; М-во освіти і науки України, Нац. ун-т «Львів. політехніка». — Львів: Вид-во Львів. політехніки, 2014. — 280 с.
5. Ковалюк Т.В. Основи програмування. – К.: Видавнича група ВНУ, 2005.

6. Караванова Т.П. Інформатика: основи алгоритмізації та програмування: 777 задач, з рекомендаціями та прикладами К.: Генеза, 2009.- 285 с.
7. В.М. Ільман. Алгоритми, дані і структури. Навч. посіб. /О.П. Іванов, Л.О. Панік. Дніпропет. нац. ун-т залізн. трансп.ім. акад. В. Лазаряна. – Дніпро, 2019. – 134 с.
8. Кузьменко І.М., Дацюк О.А. Базові алгоритми та структури даних. Навчальний посібник.К.: КПІ ім. Ігоря Сікорського, 2022. 137 с.
9. Алгоритми та методи обчислень: навчальний посібник / М. А. Новотарський. Київ, КПІ ім. Ігоря Сікорського, 2019. 407 с.

Додаткові джерела

1. Курс CS50 : https://courses.prometheus.org.ua/courses/course-v1:Prometheus+CS50+2019_T1
2. Розробка та аналіз алгоритмів:
https://courses.prometheus.org.ua/courses/KPI/Algorithms101/2015_Spring/course/
3. Робота з деревами: <http://cppstudio.com/uk/293/cat/>
4. Візуалізація алгоритмів КМП та БМ:
<http://jovilab.sinaapp.com/visualization/algorithms/strings/kmp>
<http://jovilab.sinaapp.com/visualization/algorithms/strings/boyer-moore-horspool>
5. Візуалізація деяких алгоритмів на графах, сортування та роботи з бінарними деревами: <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>

9. ДОДАТКИ

Перелік питань та завдань до модульного контролю

Модуль 1

Теоретичні питання

1. Поняття алгоритму. Походження алгоритмів.
2. Властивості алгоритмів. Способи опису алгоритмів.
3. Алгоритм Евкліда: опис, аналіз, ефективність.
4. Базові алгоритмічні структури: слідування, розгалуження, цикл з передумовою.
5. Додаткові алгоритмічні структури: вибір, цикл з післяумовою, цикл з параметром.
6. Представлення додаткових алгоритмічних структур через базові.
7. Поняття структури даних. Класифікація структур. Операції над структурами даних.
8. Класифікація простих структур даних.
9. Цілочислові беззнакові типи і їх представлення.
10. Прямий, обернений та доповняльний код.
11. Виконання операції додавання у доповняльному коді. Проблема переповнення.
12. Цілочислові знакові типи і їх представлення.
13. Нормалізація і представлення дійсних чисел.
14. Тип Float. Представлення в пам'яті.
15. Тип Double. Представлення в пам'яті.
16. Тип Extended. Представлення в пам'яті.
17. Спеціальні значення у форматах дійсних чисел.
18. Особливості комп'ютерної арифметики.
19. Двійково-десяткові типи. Представлення та операції.
20. Символьний тип. Представлення ASCII та Unicode. Операції над символьними даними.
21. Бітові типи. Представлення в пам'яті та операції на даними.
22. Логічний тип. Операції над даними.
23. Перелічувальний тип. Інтервальні типи.

Модуль 2

24. Адресація пам'яті. Види адрес.
25. Одержання фізичної адреси в реальному режимі процесора.
26. Показчики. Види показчиків.
27. Операції над показчиками.

28. Особливості статичних структур даних.
29. Множини. Фізичне представлення множин. Типи множин. Операції над множинами.

30. Вектори. Логічна структура вектора. Машинне представлення вектора. Адресація елементів вектора.
31. Послідовний (лінійний) пошук у векторі. Вдосконалення лінійного пошуку.
32. Бінарний пошук у векторі. Можливі реалізації та їх складність.
33. Постановка задачі сортування статичних структур на прикладі векторів.
34. Сортування векторів прямим вибором.
35. Сортування векторів прямою вставкою.
36. Сортування векторів прямим обміном.
37. Сортування векторів прямим злиттям.
38. Вдосконалення методу обмінів. Шейкер-сортування векторів.
39. Швидке сортування векторів.
40. Вдосконалення методу вставки. Сортування Шелла.
41. Вдосконалені методи сортування векторів злиттям.
42. Порівняння методів сортування векторів.

Типові задачі

1. Подайте наведені нижче десяткові числа у цілочислових беззнакових типах:
 а) 9 б) 1025 в) 128 г) 10127
2. Подайте у десятковій системі числення числа, записані у цілочислових беззнакових типах:
 а) 10000000 б) 10111001
 в) 00000000 00010100 г) 11111111 11111111
3. Подайте наведені нижче десяткові числа у цілочислових знакових типах:
 а) 9 б) -12 в) -128
 г) 127 д) -32768
4. Подайте у десятковій системі числення числа, записані у цілочислових знакових типах:
 а) 10000000 б) 10111001
 в) 00010100 г) 11111111 11111111
5. Розв'яжіть приклади шляхом переведення десяткових чисел у двійковий доповняльний код (довжиною 8 біт). Вкажіть на помилки переповнення, якщо вони є.
 а) -85+79 б) -98-111 в) 25+112
6. Знайдіть представлення чисел у дійсних типах (Single, Double, Extended):
 а) -13 $\frac{3}{4}$ б) 16 $\frac{5}{8}$ в) 10,1116
7. Декодуйте бітову комбінацію як число у форматі з рухомою комою:
 01000000110010000000000000000000
8. Виконайте наведені нижче логічні операції над бітовими даними:
 а) 01001011 AND 10101011 б) 01001011 OR 10101011
 в) 01001011 XOR 10101011 г) NOT 11100011
9. Знайдіть фізичну адресу у реальному режимі за значенням покажчика:
 0AB9h:00FFh.

10. Розробити програму для визначення кількості різних цифр у десятковому записі натурального числа. Використати множини та операції над ними.

11. Дано рядок символів з декількох слів, розділених пропуском. Розробити програму для визначення множини голосних, які зустрічаються: а) хоча б в одному слові, б) в кожному слові.
12. Дано цілочисловий вектор . Розробити блок-схему алгоритму для визначення кількості від'ємних значень серед елементів з парними індексами. Оцінити складність алгоритму.
13. Дано вектор і число V . Розробити блок-схему алгоритму для вилучення з вектора всіх елементів, які менші заданого значення V . Оцінити складність алгоритму.
14. Скласти блок-схему алгоритму визначення максимального елемента у кожному рядку матриці A , розмірності $m \times n$. Із знайдених елементів сформуувати вектор. Оцінити складність алгоритму.
15. Дано квадратну матрицю A . Розробити блок-схему алгоритму для визначення суми елементів, розташованих на діагоналях матриці, які проходять через елемент $A[p, k]$.
16. Дано розріджену матрицю. Побудувати її можливе представлення у пам'яті.

Модуль 3

Теоретичні питання

1. Масиви. Логічна структура та машинне представлення масиву. Операції над масивми.
2. Записи як структурний тип даних. Представлення у пам'яті. Записи з варіантами.
3. Таблиці як статичні структури. Представлення таблиць. Проблема пошуку у таблицях.
4. Поняття хешування. Досконале хешування. Типи алгоритмів хешування.
5. Хеш-функція. Вибір функції хешування. Розв'язування конфліктів при хешуванні.
6. Закрите хешування. Додавання, пошук і вилучення елементів при лінійному рехешуванні.
7. Особливості напівстатичних структур. Способи представлення.
8. Стеки. Логічна структура. Основні операції.
9. Моделювання стека за допомогою вектора. Програмна реалізація операцій над стеком.
10. Стеки в обчислювальних системах: приклади застосування та проблеми.
11. Черги FIFO. Логічна структура та машинне представлення. Черги з пріоритетами.
12. Операції над чергою та їх програмна реалізація (векторне представлення).
13. Черги в обчислювальних системах: приклади застосування та проблеми.
14. Деки, їх структура та машинне представлення. Операції над деком. Застосування деків.
15. Рядки як напівстатична структура. Особливості рядків.
16. Операції над рядками та їх реалізація у різних мовах програмування.

17. Представлення рядків у пам'яті за допомогою векторів.

18. Задача пошуку у тексті. Прямий пошук рядка.
19. Алгоритм Кнута-Морріса-Пратта.
20. Алгоритм Боуера-Мура.
21. Динамічні структури даних. Переваги та недоліки динамічних структур.
22. Класифікація лінійних зв'язних списків та їх машинне представлення.
23. Лінійні однозв'язні списки. Опис та ініціалізація.
24. Додавання елементів до лінійного однозв'язного списку. Часткові випадки.
25. Перегляд та обробка елементів лінійного однозв'язного списку. Пошук у списку.
26. Вилучення елементів із лінійного однозв'язного списку. Часткові випадки.
27. Застосування лінійних однозв'язних списків для моделювання стеку та черги.
28. Методи сортування лінійних однозв'язних списків.
29. Застосування лінійних однозв'язних списків для реалізації рядків.
30. Списки і таблиці. Спискові алгоритми відкритого хешування.
31. Представлення списків векторами покажчиків. Статичні вектори покажчиків.
32. Операції із списками, представленими статичними векторами покажчиків.

Модуль 4

33. Поняття бінарного дерева. Основні терміни. Часткові випадки бінарних дерев.
34. Опис, ініціалізація та побудова впорядкованого бінарного дерева при списковому представленні.
35. Вилучення вершини із впорядкованого бінарного дерева.
36. Пошук елемента у впорядкованому бінарному дереві.
37. Обробка і виведення дерева. Сортування за допомогою дерева.
38. Обходи бінарних дерев. Реалізація обходів.
39. Оцінка складності операцій з бінарними деревами.
40. Способи представлення бінарних дерев: використання масивів, нелінійних розгалужених списків.
41. Застосування бінарних дерев. Пошук дублікатів. Представлення арифметичних виразів
42. Проблема збалансованості впорядкованого бінарного дерева. Ідеально збалансовані дерева.
43. Сильно розгалужені дерева (дерева). Основні терміни. Представлення дерев.
44. Графи. Логічна структура.
45. Представлення графів. Способи обходу графа.
46. Алгоритми побудови кістякового дерева. Алгоритм Прима.
47. Алгоритм Крускала.
48. Алгоритми обчислення найкоротших відстаней. Алгоритм Дейкстри та Флойда
49. Пошук центру графа.

50. Пошук компонент зв'язності графу.

Типові задачі

У всіх задачах вважати, що елементами списку є цілі числа. Двох однакових значень у списку бути не може. У задачах 4 – 12 список вважати побудованим.

Розробити програму (процедуру) для:

1. побудови списку за типом стеку. Елементами є числа, що вводяться із клавіатури. Кількість чисел наперед невідома, закінчення введення – число 0;
2. побудови списку за типом черги. Елементами є числа, що вводяться із клавіатури. Кількість чисел наперед невідома, закінчення введення – число -128;
3. побудови впорядкованого списку. Елементами є числа, що вводяться із клавіатури. Кількість чисел наперед невідома, закінчення введення – число 127;
4. додавання елемента у список перед елементом з номером N (після елемента з номером N);
5. вилучення із списку елемента з номером N (елемента, який стоїть перед елементом з номером N);
6. перестановки у списку елементів із номерами K і N ;
7. копіювання K елементів списку, починаючи із позиції N , у новий список;
8. злиття двох існуючих списків у новий список;
9. пошуку у списку елемента із значенням ключового поля N ;
10. пошуку у списку максимального (мінімального) елемента;
11. впорядкування списку методом вставки (вибору);
12. формування нового списку із елементів, які одночасно належать до двох існуючих списків;
13. формування двох нових списків, у одному – додатні елементи існуючого списку, у другому – від'ємні.
14. Розробити програму для вилучення із дерева довільного елемента. Нове дерево вивести на екран.
15. Розробити програму для сортування послідовності чисел із використанням бінарного дерева.
16. Розробити алгоритм, що здійснює пошук у текстовому файлі заданих слів. Для кожного слова повинна визначатися кількість входжень і номера рядків тексту. Якщо зазначені слова у файлі відсутні, то програма повинна виводити відповідне повідомлення.
17. Дано типізований файл дійсних чисел. Розробити програму, що визначає середнє арифметичне значення компонентів файлу. Видалити з файлу всі компоненти, менші знайденого середнього арифметичного. Вивести на екран вихідний, переформований файли й значення середнього арифметичного.