| | |
|---|---|
| **Шифр та назва спеціальності** | 122 "Комп'ютерні науки" |
| **Назва освітньої програми** | Комп'ютерні науки |
| **Назва дисципліни** | **Основи програмування** |
| **Категорія дисципліни** | **Програмування, алгоритмізація** |
| **Кількість студентів (поточний рік)** | **104** |
| **Курс/Семестр** | **1 курс - 1,2 (два семестра)** |

## ЗАГАЛЬНА ІНФОРМАЦІЯ ПРО ДИСЦИПЛІНУ

| | |
|---|---|
| **Анотація** | Завданням дисципліни є набуття студентами теоретичних знань про принципи побудови сучасних мов програмування, вивчення основних конструкцій мов, типових задач алгоритмічного програмування, набуття досвіду роботи в інтегрованому середовищі розробки програм (на прикладі компілятора Visual Studio C++) та проходженні повного циклу розробки (розробка алгоритму - кодування - компіляція - відлагодження - документування). Предметом вивчення навчальної дисципліни є основи алгоритмічного мислення, а також фундаментальні підходи до процедурного та об’єктно-орієнтованого програмування. |
| **Мета навчальної дисципліни** | Метою курсу є ознайомлення з основними середовищами та технологіями програмування, основами створення програмних продуктів, вивчення основних понять процедурного та об’єктно-орієнтованого програмування, а також базових понять алгоритмічного програмування, методів аналізу та реалізації найпростіших алгоритмів засобами мови програмування C++. |
| **Типи занять та контролю** | Лекції, лабораторні роботи, практичні заняття, самостійна робота. Підсумковий контроль – іспит |

| **Загальний обсяг (кредитів)** | | **Лекції (занять)** | **64** | **Лабораторні (занять)** | **64** | **Практичні (занять)** | **64** | **Самостійна (годин)** | **168** |
|---|---|---|---|---|---|---|---|---|---|

| | |
|---|---|
| **Попередні дисципліни** | Вступ до комп'ютерних наук та інформаційних технологій, Математичний аналіз |
| **Необхідні знання та вміння** | Базові навички роботи з комп'ютером. Вміння логічно мислити, знання сучасного математичного апарату неперервного та дискретного аналізу, лінійної алгебри, аналітичної геометрії для розв’язання задач теоретичного та прикладного характеру. Вміння проектувати, розробляти та аналізувати алгоритми розв’язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів. Вміння вибирати та застосовувати парадигму, технології та мови програмування. |

## МАТЕРІАЛЬНО-ТЕХНІЧНЕ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДИСЦИПЛІНИ

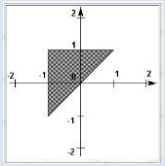
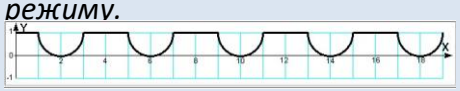
| | |
|---|---|
| **Наявні у навчальних аудиторіях у ЗВО** | Комп'ютери з ОС Windows (або інше),Visual Studio (або інше), доступ до Internet. |
| **Необхідні для самостійної роботи студента** | Ті ж вимоги, що і для навчання в аудиторії |

## СТРУКТУРА ТА ЗМІСТ ДИСЦИПЛІНИ

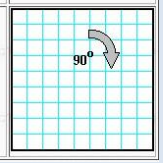
| № | Теоретична складова<br>*Назва, перелік питань або анотація лекції* | Годин | Практична складова<br>*Опис та приклад завдання, а також посилання на методичні матеріали* | Годин | Інструменти, засоби, та технології |
|---|---|---|---|---|---|
| **Змістовий модуль 1 - Системи числення** | | | | | |
| **1** | Тема 1. Арифметичні та логічні основи комп’ютера<br>Мета та завдання дисципліни. Загальні відомості про дисципліну, її зв'язок з іншими дисциплінами. Поняття про системи числення (позиційну та непозиційну). Перетворення двійкових, вісімкових, десяткових та шістнадцяткових чисел в інші системи числення. Арифметичні дії над двійковими числами. | 2 | Практичне заняття. Системи числення. Переклад з однієї системи числення в іншу. | 2 | Комп'ютер (ноутбук) для перевірки онлайн |
| **Змістовий модуль 2 - Введення в розробку і кодування алгоритмів. Основні поняття мови програмування. Базові типи даних** | | | | | |
| **2** | Тема 2. Аналіз алгоритмів програм.<br>Основні алгоритмічні блоки. Алгоритми та програми. Блок-схема як засіб графічного зображення алгоритмів. Лінійні алгоритми. Алгоритми з розгалуженням. Цикли та циклічні структури. Ємнісна складність алгоритмів. Часова складність алгоритмів. Алгоритми логарифмічної, лінійної, поліноміальної та експоненційної складності. | 2 | Практичне заняття. Лінійні алгоритми. Алгоритми з розгалуженням. Алгоритми циклічних структур. | 2 | |
| **3** | Тема 3. Введення в програмування.<br>Введення до програмування. Види ПЗ. Етапи розробки ПЗ. Структура програми, літерали, синтаксис мови, змінні та константи, сфера їх дії. Інтегроване середовище редагування, налагодження, компіляції та створення програм. Найменування змінних, обмеження на використання певних символів. Типи даних, їх розміри. Ключові слова. Константи. Оголошення змінних. | 2 | Лабораторна робота. Робота в системі програмування Visual Studio. Налагодження і тестування програми. Дослідження внутримашинного подання примітивних типів по дампах пам'яті. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **4** | Тема 4. Базові типи даних | 2 | | 2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

| | | | | | |
|---|---|---|---|---|---|
| | Цілі та дійсні числа. Символьні дані, коди та алгоритми кодування. Логічні дані. Адресні дані та адресна арифметика. Фізичний рівень подання базових типів даних, системи адресації. Поняття типізації у мовах програмування. | | Практичне заняття. Базові типи даних у мовах програмування. Основні операції над даними. Арифметичні операції. Логічні та побітові операції.<br><br>Лабораторна робота. Створення лінійних програм. Робота з точками зупинки. Приклад простої програми з арифметичними операціями над даними. | | |
| **5** | Тема 5. Введення/виведення в мові С.<br><br>Структура програми. Область дії змінних. Перетворення типів. Введення/виведення в мові С. Стандартні функції введення/виведення мов С та С++. Використання коментарів. Директиви препроцесора. | 2 | Практичне заняття. Базові типи даних та введення/виведення.<br><br>Лабораторна робота. Базові типи даних та введення/виведення.<br>Приклади завдань:<br>*Складіть програму, яка підраховує і виводить значення t1 і t2 за формулами, які наведені у Вашому варіанті індивідуального завдання. Визначте області допустимих значень параметрів формул і задайте довільні значення з цих областей.* | 2<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **6** | Тема 6. Основні операції в С/С++<br><br>Таблиця пріоритетів. Операції присвоєння. Повна та скорочена форма операції присвоєння. Операції інкремента та декремента. Арифметичні операції та математичні функції мови С/С++. Логічні та побітові операції. Пріоритет виконання операцій. | 2 | Лабораторна робота. Арифметичні операції та математичні функції мови С++. Побітові операції.<br>Приклади завдань:<br>*Скласти дві програми, перша з яких вводить складові частини структури даних, наведеної у Вашому варіанті індивідуального завдання, і формує з них задану упаковану структуру. Друга програма вводить упаковану структуру як 16-ковий число і виводить значення окремих її складових частин.*<br><br>[illegible] | 2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

| [illegible] разряда | 15 | 14 | 13 | 12 | 11 | 10 | 09 | 08 | 07 | 06 | 05 | 04 | 03 | 02 | 01 | 00 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Значение | H | H | T | T | T | T | T | T | T | T | T | S | S | S | S | S |

| Змістовий модуль 3 - Оператори управління програмою | | | | | |
|---|---|---|---|---|---|
| **7** | Тема 7. Умовні оператори<br><br>Структури управління. Структура вибору if. Структура вибору if/else. Обробка складних умов в мові C/C++. Багатоваріантний вибір switch. Приклади програм з реалізованими структурами управління. | 2 | Практичне заняття. Умовний оператор та Багатоваріантний вибір.<br><br>Лабораторна робота. Умовний оператор в мові C++.<br>Приклади завдань:<br>*Побудувати програму, яка вводить координати точки (x, y) і визначає, чи потрапляє точка в заштриховану область на малюнку, який відповідає Вашому варіанту. Попадання на кордон області вважати попаданням в область.*<br>  | 4<br><br>2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **8** | Тема 8. Оператори циклу<br><br>Три види операторів циклу. Вкладені цикли. Структура повторення for. Блок-схема алгоритму структури for. Синтаксис структури for. Приклади структур for. Структура циклу з передумовою while. Структура циклу з післяумовою do/while. Рекомендації щодо вибору структур повторення в залежності від алгоритму. Приклади програм з реалізованими структурами повторення. Оператори break і continue. | 2 | Практичне заняття. Оператор циклу for, while та do-while.<br><br>Лабораторна робота. Оператор циклу for, while та do-while.<br>Приклади завдань:<br>*1. Для ряду, члени якого обчислюються за формулою, що відповідає Вашим індивідуальним завданням, підрахувати суму членів ряду з точністю до 0.000001 і суму перших 10 членів ряду. Якщо Ви вважаєте це за необхідне, можете спростити або перетворити вираз.*<br>*2. Для функції Y = f (X), графік якої наведено в Вашому індивідуальному завданні, вивести на екран значення Y для X = 0, 0.25, 0.5, ... 19.75. Не обов'язкове для виконання додаткова вимога - отримати на екрані графік функції за допомогою текстового режиму.*<br>  | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

**Змістовий модуль 4 - Статичні одновимірні та багатовимірні масиви**

| | | | | | |
|---|---|---|---|---|---|
| **9** | Тема 9. Статичні масиви<br><br>Статичні одновимірні та багатовимірні масиви. Алгоритми упорядкування масивів. Лінійний пошук, бінарний пошук. Стратегії упорядкування. | 4 | Практичне заняття. Алгоритми роботи з послідовностями. Сортування одновимірних масивів. Алгоритми пошуку в одновимірних масивах. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **10** | Тема 10. Робота з одновимірними масивами.<br><br>Масиви як структурований тип даних. Визначення масивів. Синтаксис об'яви масивів. Поняття “елемент масиву” та “індекс масиву”. Синтаксис використання масивів. Алгоритми роботи з послідовностями. Сортування одновимірних масивів. Алгоритми пошуку в одновимірних масивах. | 4 | Лабораторна робота. Алгоритми роботи з послідовностями. Робота з одновимірними масивами.<br>Приклади завдань:<br>*Оголосити масив цілих чисел і заповнити його випадковими значеннями. Розмір масиву і діапазон значень його елементів задані в Вашому варіанті індивідуального завдання. В індивідуальних завданнях зазначено також, яку обробку масиву слід зробити.*<br><table><tr><th>№ варианта</th><th>Размерность массива</th><th>Диапазон значений</th><th>Что нужно сделать</th></tr><tr><td>1</td><td>50</td><td>-100 - 100</td><td>Заменить все элементы с отрицательным значением на значение минимального не равного 0 положительного элемента</td></tr></table> | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **11** | Тема 11. Робота з багатовимірними масивами.<br><br>Приклади програм з багатовимірними масивами. Особливості реалізації двовимірних масивів. Зв’язок між лінійним та багатовимірним індексами.<br><br>Реалізація алгоритмів, пов’язаних з обробкою матриць. Алгоритми перетворення матриць. Алгоритми заповнення, копіювання, відображення, повороту в двовимірних масивах | 4 | Практичне заняття. Алгоритми роботи з матрицями. Робота з багатовимірними масивами.<br><br>Лабораторна робота. Алгоритми роботи з матрицями. Робота з багатовимірними масивами.<br>Приклади завдань:<br>*Створити квадратну матрицю цілих чисел розміром 9x9. В індивідуальних завданнях зазначено, яку обробку матриці потрібно виконати.*<br><table><tr><th>№ варианта</th><th>Содержание задания</th><th>Иллюстрация</th></tr><tr><td>1</td><td>Заполнить матрицу случайными числами. Развернуть матрицу на 90° по часовой стрелке.</td><td>  </td></tr></table> | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

**Змістовий модуль 5 - Функції і управління пам’яттю**

| | | | | | |
|---|---|---|---|---|---|
| **12** | Тема 12. Загальні поняття щодо структурного програмування<br>Програмні модулі мови С. Функції математичної бібліотеки. Функції бібліотеки стандартного введення/виведення. Використання функцій при розробці програм на мові С. Визначення функцій. Прототипи функцій та файли заголовків.<br><br>Розробка власних функцій. Виклик функції за значенням. Способи обміну інформацією між функціями. Локальні та глобальні змінні. | 2 | Практичне заняття. Глобальні та локальні змінні. Області видимості змінних. Прототипи функцій. Розробка власних функцій. Функції користувача.<br><br>Лабораторна робота. Створення функцій користувача.<br>Приклади завдань:<br>*Скласти програму, яка вирішує завдання для лабораторної роботи по багатовимірним масивом з такими додатковими умовами: розмірність матриці повинна вводитися при виконанні програми; саме рішення задачі повинно бути оформлено у вигляді* функції, якій передається матриця і її розмірність. | 4<br><br>2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **13** | Тема 13. Функції користувача в мові С/С++.<br>Визначення функції. Внутрішні та зовнішні змінні. Значення, що повертає функція. Розробка програми у вигляді проекту. Зовнішні змінні. Використання С-препроцесора. Включення додаткових файлів, макропідстановки, умовна компіляція. | 4 | Лабораторна робота. Конструювання програм із декількох файлів. Побудова проектів. Створення бібліотек. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **14** | Тема 14. Рекурсія.<br><br>Приклади використання рекурсії. Реалізація стандартних алгоритмів обробки масивів за допомогою рекурсії. | 4 | Практичне заняття. Реалізація стандартних алгоритмів обробки масивів за допомогою рекурсії.<br><br>Лабораторна робота. Функції. Передача масивів як параметр функції. Рекурсія.<br>Приклади завдань:<br>*1. Ввести послідовність чисел (закінчення введення - 0) і вивести їх в зворотному порядку.*<br>*2. Написати рекурсивну програму друку цифр цілого позитивного числа N.* | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

**Змістовий модуль 6 - Структури і масиви структур**

| | | | | | |
|---|---|---|---|---|---|
| **15** | Тема 15. Робота зі структурами<br><br>Створення власних типів даних за допомогою структур. Об’ява структур. Об’ява структурних змінних. Доступ до елементів структури. Використання масивів структур. Використання структур з функціями. | 2 | Практичне заняття. Структури. Принципи об'явлення та використання структур.<br><br>Лабораторна робота. Структури. Масиви структур.<br>Приклади завдань:<br>*Скласти програму, в якій будуть вводитися 7 - 10 рядків таблиці, зразок якої наведено в Вашем варіанті індивідуального завдання і виводиться на екран таблиця - відразу ж після введення і після сортування її за значеннями в першому стовпці.*<br>**Вариант №1**<br>Некоторые виды антилоп<br>Название \| Группа \| Место обитания \| Численность популяции<br>Джейран \| А \| Азия \| 30000<br>Гну \| В \| Африка \| 560000<br>Бейза \| Н \| Африка \| 2500<br>Групи: А - настоящие антилопы, В - коровьи антилопы, Н - лошадиные антилопи | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше)) |

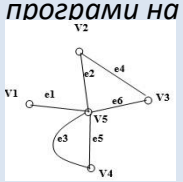
**Змістовий модуль 7 - Покажчики та розподілення пам'яті**

| | | | | | |
|---|---|---|---|---|---|
| **16** | Тема 16. Покажчики. Динамічна пам'ять.<br><br>Моделі пам’яті у С та їх особливості. Модель процесу виконання програми, адресний простір процесу; його поділ на частини (код, постійно існуючі дані, динамічна пам’ять, стекова, специфікатор static). Типізовані покажчики. Операції взяття адреси, розіменування покажчика. Динамічні масиви. Властивості динамічних масивів. Покажчики на елементи масивів. Адресна арифметика. Масиви як параметри функцій. Кваліфікатор const. Покажчики на структури. Використання покажчиків як аргументів функцій.<br>Покажчик на функцію. | 2 | Практичне заняття.Покажчики. Динамічна пам'ять. Покажчики та масиви.<br><br>Лабораторна робота. Покажчики та масиви. Виклик функції за посиланням. Передача параметрів за адресою.<br>Приклади завдань:<br>*Розробити програму з використанням покажчиків.*<br>*1. Згенерувати одновимірний масив. Переписати в новий масив значення, які більше введеного з екрану числа. Замінити всі негативні числа на 100, а позитивні на 200.*<br>*2. Відсортувати цілочисельний масив по зростанню. Переписати в новий масив всі позитивні елементи.* | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

**Змістовий модуль 8 - С-рядки**

| | | | | | |
|---|---|---|---|---|---|
| **17** | Тема 17. Поняття С-рядків.<br><br>Логічна структура, подання, операції, алгоритми, засоби обробки рядків у мові програмування С. | 2 | Практичне заняття.Функції роботи з рядками. Вирішення прикладних задач із застосуванням рядків. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **18** | Тема 18. Робота з рядками в С/С++.<br><br>С-рядки, особливості масивів символів. Функції роботи з рядками. Покажчики, символьні рядки, функції. Зображення рядків у програмах. Тип рядків string. Стандартні бібліотечні засоби обробки строк типу string. Вирішення прикладних задач із застосуванням рядків. | 2 | Лабораторна робота. Покажчики, символьні рядки, функції роботи з рядками.<br>Приклади завдань:<br>*Розробити функцію, яка виконує ту обробку символьного рядка, яка визначена у Вашому індивідуальному завданні. При реалізації функції забороняється користуватися функціями бібліотек мови С.*<br>**Вариант 1**<br><br>Функция подсчитывает количество слов в строке. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **Змістовий модуль 9 - Файлова організація даних** | | | | | |
| **19** | Тема 19. Вступ до організації зовнішніх даних.<br><br>Файлові структури. Фізичне та логічне подання даних на зовнішній пам'яті. Файл як тип даних. Файл у програмі та в операційній системі. Операції над файлами та над даними файлів. Файлові системи. Системні каталоги. | 2 | Практичне заняття. Операції над файлами та над даними файлів.<br><br>Лабораторна робота. Форматне файлове введення-виведення.<br>Приклади завдань:<br>*Використовуючи напрацювання лабораторно роботи за темою "Структури" реалізувати роботу зі структурою за наступним меню:*<br>*1. Введення з екрану і запис в файл.*<br>*2. Введення випадковим чином і запис в файл.*<br>*3. Додати запис в початок файлу.*<br>*4. Додати запис в кінець файлу.*<br>*5. Друк однієї записи з файлу за номером.*<br>*6. Друк всіх записів з файлу.*<br>*7. Вихід з програми.* | 4<br><br>2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше)) |

| | | | | | |
|---|---|---|---|---|---|
| **20** | Тема 20. Файлове введення/виведення.<br><br>Стандартне та пряме виведення. Загальні уявлення про файлову систему. Ієрархія даних. Файли і потоки. Стандартні методи для роботи з файлами. Створення файлів послідовного доступу. Читання з файлів послідовного доступу. Приклади програм. | 2 | Практичне заняття. Створення файлів послідовного доступу. Читання із файлів послідовного доступу.<br><br>Лабораторна робота. Пряме файлове введення-виведення.<br>Приклади завдань:<br>*1. Записати в файл (f1) масив (mas1) 10 чисел. Переписати з (f1) в інший масив (mas2) і в інший файл (f2). Роздрукувати (mas2). З файлу (f2) переписати дані послідовно в змінну x і роздрукувати (fprintf (), fscanf ()).*<br>*2. Записати в файл (f1) весь масив (mas1) 10 чисел. Вважати поелементно в інший масив і знайти суму парних (fread (), fwrite ()).* | 4<br><br>2 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **Змістовий модуль 10 - Односпрямовані та двоспрямовані списки** | | | | | |
| **21** | Тема 21. Списки<br><br>Подання у пам'яті. Одно- та двоспрямовані списки. Організація лінійних списків. Вирішення прикладних задач із застосуванням лінійних списків. | 4 | Практичне заняття. Організація лінійних списків. Вирішення прикладних задач із застосуванням лінійних списків.<br><br>Лабораторна робота. Реалізація операцій обробки лінійних списків. Реалізація операцій обробки циклічних списків.<br>Приклади завдань:<br>*Використовуючи напрацювання лабораторної роботи за темою "Структури" організувати роботу зі списком за наступним меню: 1. Створення списку. 2. Друк списку з початку або з кінця. 3. Пошук за критерієм (здійснити вибір критерію). 4. Додавання елемента (в будь-який заданий місце). 5. Видалення будь-якого елементу. 6. Сортування. 7. Вихід з програми.* | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **Змістовий модуль 11 - Нелінійні СД** | | | | | |

| | | | | | |
|---|---|---|---|---|---|
| **22** | Тема 22. Графи: подання у ЕОМ, операції, алгоритми, використання у програмних та інформаційних системах. Класифікація. Поняття дерева та лісу. Орієнтовані, упорядковані та бінарні дерева. Подання у ЕОМ - основні та альтернативні методи. Операції, алгоритми, використання у програмних та інформаційних системах.<br>Орієнтовані та неорієнтовані графи.<br>Основні визначення. Види графів та операції над графами. Представлення графів за допомогою списків суміжності та масиву ребер. | 4 | Практичне заняття. Представлення орієнтованих та неорієнтованих графів за допомогою матриці суміжності. та інцидентності.<br><br>Лабораторна робота. Матричні способи представлення графів.<br>Приклади завдань:<br>*Розробити алгоритм програми, що реалізує переклад з заданого способу матричного представлення графа в інший, з огляду на при цьому вихідний тип графа (неорієнтований, орієнтований). 2. Скласти програму, що реалізовує переклад з заданого способу матричного представлення графа в інший. Передбачити введення вихідних даних з файлу і виведення результатів роботи програми на екран.*<br>  | 4<br><br>4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше)) |
| **Змістовий модуль 12 - Основи об’єктно-орієнтованого програмування** | | | | | |
| **23** | Тема 23. Загальні положення мови С++<br><br>Основні принципи ООП. Ключові слова С++. Різниця між С та С++. | 2 | Практичне заняття.Використання класів. Конструк об’єктів класу. Передача об’єктів в функції. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |
| **24** | Тема 24. Введення в класи. Конструктори та деструк<br><br>Конструктори за умовчанням. Конструктори з параметрами. Конструктор копіювання. Покажчик this. Функції, що вбудовуються (inline-функції). Робота з простими класами. Покажчики на об’єкти. Класи, структури, об’єднання. Різниця між класами та структурами. Масиви об’єктів класу. Передача об’єктів в функції. Повертання об’єктів із функцій. Посилання. Передавання посилань на об’єкти. | 4 | Лабораторна робота. Прості класи. Доступ до елементів класу. Масиви об’єктів класу. Передача об’єктів в функції. | 4 | Комп'ютер (ноутбук) з встановленою Visual Studio (або інше) |

## ТЕМИ ТА ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

| № | Назва та опис завдання | Методи контролю та критерії оцінювання | Годин |
|---|---|---|---|
| 1 | **Проріджені масиви. Упорядкування розподіленням. Упорядкування злиянням. Таблиці прямого доступу.** | Оцінка роботи програмної реалізації | 12 |
| 2 | Стеки, черги, деки. Використання у програмних та інформаційних системах. Організація черг та стеків. Вирішення прикладних задач із застосуванням стеків. | Оцінка роботи програмної реалізації | 12 |
| 3 | Циклічні списки. Використання списків у програмних та інформаційних системах. Текстові процесори. | Оцінка роботи програмної реалізації | 12 |
| 4 | Алгоритми на графах. Алгоритм Дейкстри. Пошук найкоротшого шляху між парами вершин. Остовні дерева мінімальної вартості. Алгоритм Флойда. Порівняння алгоритмів Флойда та Дейкстри. Пошук у глибину. Пошук у ширину. | Оцінка роботи програмної реалізації | 12 |
| 5 | Алгоритми обробки деревоподібних структур. Базові визначення. Впорядковані та невпорядковані дерева. Прямий, зворотний та симетричний обходи дерев. Бінарні дерева. Представлення бінарних дерев. Алгоритми побудови бінарного дерева та пошуку в бінарних деревах. | Оцінка роботи програмної реалізації | 12 |
| 6 | Опрацювання лекційного матеріалу | Оцінки (бали) за контрольні роботи | 16 |
| 7 | Підготовка до практичних та лабораторних занять | Оцінки (бали) за практичні роботи та оцінка роботи програмної реалізації на лабораторних роботах | 64 |
| 8 | Виконання індивідуальних завдань (РЗ та КР) | Оцінка роботи програмної реалізації | 40 |

## ПРОЕКТ (за наявністю)

| № | Назва та опис завдання | Метод контролю та захисту | Строки виконання |
|---|---|---|---|
| 1 | Розрахункове завдання (1 семестр). Порівняльний аналіз методів сортувань.<br>Вивчити методи сортування, в звіті привести докладний опис методів. Програмно реалізувати методи сортування масивів. Розробити і програмно реалізувати засіб для проведення експериментів по визначенню часових характеристик алгоритмів сортування. Побудувати графік залежності часу виконання алгоритму сортування від кількості елементів у відсортованому масиві при сортуванні упорядкованих, неупорядкованих і упорядкованих в зворотному порядку масивів. | Оцінка роботи програмної реалізації, оцінка наведених в звіті результатів порівняльного аналізу різних методів сортування. | 15-16 тиждень |
| 2 | Курсова робота (2 семестр). Вивчення технології візуального проектування і подієвого програмування. Створення діалогового вікна і написання функцій обробки подій. Створення програми за допомогою призначеного для користувача інтерфейсу для Windows Form. | Оцінка роботи програмної реалізації, оцінка оформленої пояснювальної записки. | 37-38 тиждень |

# РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ ТА НАВЧАЛЬНІ МАТЕРІАЛИ

## Основна

| № | Назва | До теми |
|---|---|---|
| 1 | 1. Herbert Schildt. C: The Complete Reference (Osborne Complete Reference Series). – McGraw Hill; 4th edition (May 17, 2000). ISBN-13 : 978-0072121247. – 1008 p. | 1-12 |
| 2 | 2. Bjarne Stroustrup. The C++ Programming Language. – Addison-Wesley Professional; 4th edition (May 9, 2013). ISBN-13 : 978-0275967307 – 1376 p. | 1-12 |
| 3 | 3. Stephen Prata, C Primer Plus, Fifth Edition. ISBN: 0-672-32696-5. – 2014. – 512 c. | 1-12 |
| 4 | 4. Васильєв О. Програмування С++ в прикладах і задачах. – Київ, Ліра-К, 2019, 382 с. | 1-12 |
| 5 | 5. Саттер Г. Вирішення складних задач на С++. Вільямс, 2015.400 с. | 1-12 |
| 6 | 6. Stanley Lippman. C++ Primer. – Addison-Wesley Professional; 5th edition (August 6, 2012). ISBN-13 : 978-0321714114. – 976 p. | 1-12 |
| 7 | 7. Джордж Хайнеман, Гері Полліс, Стенлі Селков. Алгоритми. Довідник з прикладами на С, С++. - Діалектика, 2017. 432 с. | 1-12 |

## Додаткова

| № | Назва | До теми |
|---|---|---|
| 1 | 1. Любченко Н. Ю., Соболь М.О., Паржин Ю.В., Пугачов Р.В. Основи програмування на С/С++ в прикладах. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2021. – 120 с. | 1-12 |
| 2 | 2. Любченко Н. Ю., Соболь М.О., Паржин Ю.В., Пугачов Р.В., Ивашко А.В. Основи програмування на С/С++ в прикладах. Частина 2. / Навчально-методичний посібник. – Харків: НТУ "ХПІ". – 2022. – 200 с. | 1-12 |
| 3 | 3. Уроки программування на С++ : [сайт]. URL: https://acode.com.ua/urok-1-vvedennya-v-programuvannya/ | 1-12 |
| 4 | 4. International Standard ISO/IEC 14882:2014(E) – Programming Language C++ , ISBN-13: 978-0321563842: [сайт]. URL: https://isocpp.org/std/the-standard. | 1-12 |

# КОНТРОЛЬНІ ЗАХОДИ

| № | Назва та опис | Методи контролю та критерії оцінювання |
|---|---|---|
| 1 | **Тема 1** | Контрольна робота "Системи числення", " Подання двійкових чисел в прямому, зворотному і додатковому кодах. Арифметичні дії з числами в різних системах числення" |
| 2 | **Тема 2 - Тема 24** | За результатами захисту лабораторних робіт |
| 3 | Контрольна робот №1 | Модульна контрольна робота 1, тести після змістовного модуля №3 (1 семестр) |
| 4 | Контрольна робот №2 | Модульна контрольна робота 2, тести після змістовного модуля №5 (1 семестр) |
| 5 | Контрольна робот №3 | Модульна контрольна робота 3, тести після змістовного модуля №8 (2 семестр) |
| 6 | Контрольна робот №4 | Модульна контрольна робота 4, тести після змістовного модуля №12 (2 семестр) |
| 7 | РГЗ (1 семестр), КР (2 семестр) | За результатами виконання завдань для самостійної роботи. |
| 8 | Іспит (1 та 2 семестр) | За результатами співбесіди |

| РЕЗУЛЬТАТИ НАВЧАННЯ, ЗНАННЯ ТА ВМІННЯ | |
|---|---|
| **Результати навчання** | Вміти використовувати сучасний математичний апарат неперервного та дискретного аналізу, лінійної алгебри, аналітичної геометрії, в професійній діяльності для розв’язання задач теоретичного та прикладного характеру. Вміти проектувати, розробляти та аналізувати алгоритми розв’язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей. Вміти вибирати та застосовувати парадигму, технології та мови програмування (з позицій зручності та якості застосування) для реалізації методів та алгоритмів розв’язання прикладних задач, володіти навичками компонентного та крос-платформного програмування. |
| **Набуті знання та вміння** | Здатність до логічного мислення, побудови логічних висновків, використання формальних мов, проектування, розроблення й аналізу структур даних і алгоритмів, оцінювання їх ефективності та складності. Здатність використовувати технології проектування програмного забезпечення та мови програмування для вирішення задач обробки даних. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об’єктно-орієнтованого, функціонального, логічного, компонентного та крос-платформного програмування. |