**Загальні вимоги до програми, що розробляється, в рамках курсової роботи**

В текст програми обов'язково до­даються змістовні коментарі.

Вимоги до функціональної частини програми:

- швидкий та зручний запуск програми на виконання та коректність завершення за умови відсутності необхідних ресурсів;

- обробка виняткових ситуацій під час виконання програми;

- стійкість до введення непередбачених даних (невідповідність типів, помилкове натиснення кнопок на клавіатурі і т.д.);

- гнучкість в роботі (легке управління переходом до інших точок ви­конання програми, наприклад, по меню);

- стабільна робота програми після коректного запуску (без зависань, злетів, виведення повідомлень невідомого характеру та змісту тощо);

- правильність розрахункового апарату (порівняння з еталонними розрахунками та з тими, що отримані іншими методами);

- можливість зчитування та збереження даних, якими оперує про­грама (текстові та типізовані файли, бази даних);

- виконання програми без бібліотек середовища програмування.

Вимоги до інтерфейсної частини програми:

- дружність інтерфейсу (лаконічні та інформативні повідомлення), що прискорює розуміння правил користування програмою;

- мінімізація об'єктів управління програмою (форми, кнопки і т.д.);

- відображення усіх даних із назвою, знаком та у зручному форматі;

- наочність, повна інформативність та візуалізація отриманих ре­зультатів (графіки, діаграми, таблиці тощо);

- можливість використання елементів допомоги (НЕLР-файли тощо).

**Структура пояснювальної записки**

**1. Титульний лист.**

Міс­тить наступні складові (у такій послідовності):

- назву міністерства та вищого навчального закладу, де виконано ку­рсову роботу;

- повну тему курсової роботи;

- тип роботи:Курсова робота**,** з зазначенням назви дисципліни «Основи програмування. Фундаментальні структури даних»;

- прізвище, ініціали та номер залікової книжки автора (виконавця) роботи;

- прізвище та ініціаликерівника курсової роботи;

- місце для підписів виконавця й керівника;

- місто і рік виконання роботи.

**2. Завдання на курсову роботу**

Лист завдання на курсову роботу є другою сторінкою пояснювальної записки. У ньому зазначаються наступні позиції:

- курс, семестр та група виконавця курсової роботи;

- назва дисципліни;

- прізвище, ім'я та по батькові виконавця курсової роботи;

- повна назва теми курсової роботи;

- термін здачі студентом закінченої курсової роботи;

- вихідні (початкові) дані до розробки;

- зміст пояснювальної записки (перелік питань, які підлягають роз­робці);

- перелік графічного матеріалу;

- дата видачі завдання на курсову роботу;

- календарний план-графік виконання курсової роботи;

- підписи виконавця та керівника.

Загальні пункти завдання заповнюються кожним студентом самостійно. Решта пунктів (тема роботи, вихідні дані) узгоджуються за безпосеред­ньою участю керівника курсової роботи.

Тема заповнюється згідно узгодженого завдання . У вихідних даних відзначаються:

- умови, в яких має проектуватись та працювати програма (мова та середовище програмування, операційна система та цільова платформа);

- кількісні (та) або якісні характеристики програми після її розробки в курсовій роботі.

Перелік питань та наявність ілюстративного матеріалу в курсовій ро­боті узгоджуються із керівником.

На базі завдання розробляється календарний план-графік виконання курсової роботи.

У графі строк виконання етапів роботи мають бути прописані кін­цеві дати завершення кожного етапу виконання курсової роботи.

Лист завдання підписується студентом та затверджується керівником.

**3. Анотація**

Стислий виклад змісту документа, що містить основні відомості і ви­сновки, необхідні для початкового ознайомлення з документом. Обсяг - не більше половини сторінки тексту (5-7 речень).

**4. Зміст**

Містить у собі всі структурні частини документа крім анотації і само­го змісту з вказівкою порядкового номера, та номера першої сторінки на якій розміщується відповідна частина документу.

**5. Вступ**

У цьому розділі обґрунтовують актуальність теми; вказують мету ро­боти, розкривають найменування і призначення та стислу характеристику області застосування програми, що розробляється в курсовій роботі. Також у вступі необхідно описати сукупність алгоритмічних методів, технічних і програмних засобів, що мають місце при розробці курсової роботи.

Вступ доцільно писати після завершення роботи над основною ча­стиною.

**Основна частина пояснювальної записки**

**1. Постановка задачі**

В цьому розділі наводяться результати уточнення завдання до курсо­вої роботи, якщо таке уточнення виконувалось.

**2. Математична модель та структура програми**

Розділ містить огляд та аналіз існуючих рішень й методів розв'язан­ня задачі. За результатами огляду обирається (розробляється) і обґрунтову­ється вибір конкретного методу(ів) розв'язання задачі та формується мате­матична модель й загальна структура програми.

Результати розробки подаються у вигляді схеми програми та схеми даних. Алгоритми розв'язання задачі описуються на псевдокоді та/або символами схем алгоритмів відповідно до стандарту.

**3. Перелік і призначення режимів та структура діалогу**

В цьому розділі наводяться результати розробки структури діалогу та проектування інтерфейсу програми.

За результатами проектування діалогу визначаються:

- перелік, призначення та опис режимів роботи програми;

- структура діалогу програми (умови, порядок та засоби переходу від однієї точки діалогу до іншої);

- склад та структура меню користувача, за допомогою якого вико­нується виклик потрібних режимів;

- формат виведення на екран повідомлень (коментарів), запитів та результатів роботи програми;

- засоби виклику допомоги та довідкової інформації, що стосуються призначення режимів та порядку роботи з програмою.

**4. Структура даних та ресурсів програми**

В цьому розділі наводяться результати вибору (або розробки) струк­тур та ресурсів даних (інформаційного забезпечення) з якими працює програма та їхню характеристику.

У випадку розробки класів, для кожного класу необхідно навести:

- опис та призначення класу;

- перелік, ідентифікатори, типи та призначення компонентних даних;

- перелік, описи та призначення компонентних функцій класу;

- схему родинних зв'язків між різними класами, якщо такі зв'язки мають місце.

Для глобальних змінних програми необхідно навести:

- характер (вхідні, вихідні або допоміжні та тимчасові);

- ідентифікатор;

- вид структури даних (проста змінна, масив, структура тощо);

- тип змінної (це може бути і тип розроблений програмістом, напри­клад клас);

- смисловий зміст (призначення).

Якщо в програмі використовуються файли, то для кожного з них не­обхідно навести наступне:

- назва та призначення файлу, його тип (текстовий або двійковий);

- характер використання (тільки на запис, тільки на зчитування або і на запис і на зчитування);

- структура інформації, яка зберігається в файлі.

Якщо в програмі використовуються бази даних або інші зовнішні ре­сурси, то для кожного з цих елементів наводиться детальний опис.

**5. Опис програми**

В цьому документі наводяться результати алгоритмізації у вигляді опису логічної та функціональної структур програми. Зміст роз­ділу має включати наступні відомості.

Модульна структура програми

Наводиться довільний граф, який визначає порядок взаємодії функ­цій програми та стислий опис усіх розроблених функцій, які використову­ються в програмі (в тому числі і компонентних функцій класів):

Назва, призначення та опис функції й типів її параметрів;

Характеристика значення, яке повертається функцією (якщо є);

Коротка характеристика особливостей роботи функції. Зв'язок програми розробника зі сторонніми програмами (якщо він є)

Детально описується взаємодія програми зі сторонніми програмами. Виклик і завантаження

Вказуються засіб та процедура виклику програми з відповідного носія даних та вхідні точки в програму.

Вхідні дані

Вказуються організація, попередня підготовка, формат, опис та засіб кодування вхідних даних (також дані, які вводяться з клавіатури/файлу). Вихідні дані

Характер й організація вихідних даних (у тому числі дані, які виво­дяться на екран дисплея або файл) та формат їх виведення.

Опис всіх повідомлень

Опис повідомлень, що видаються програмою в ході виконання, опис їхнього змісту і дії, які виконуються у випадку появи таких повідомлень.

У додаток до опису програми допускається включати різноманітні ві­домості, які доповнюють цей розділ.

**6. Інструкція програміста**

Зміст інструкції програміста повинна включати наступну інформацію.

Загальні відомості, позначення й найменування програми.

Умови проектування програми:

- перелік програмного забезпечення, що необхідне для проектування програми: зовнішні бібліотеки, драйвера, шаблони, текстури тощо;

- мови та середовища програмування; компілятори, емулятори;

- тип та характеристики платформи (конфігурація ПК, операційна система), на якій проектується програма;

- периферійні пристрої та інше, що необхідне для проектування. Керівництво розробника програми

Наводиться детальний опис основних етапів проектування програми:

- встановлення, запуск та налаштування середовища програмування;

- створення каркасу проекту, додавання файлів та функцій;

- розробка ресурсів, інтерфейсу, класів та методів, змінних;

- підключення баз (якщо є), драйверів, інших ресурсів тощо;

- реалізація математичного алгоритму;

- налаштування компілятора та отримання виконавчого файлу.

Опис всіх повідомлень, що видаються розробнику в ході створення про­грами, опис їхнього змісту і дії, які необхідно виконати у випадку появи цих повідомлень.

Оформлення інсталяції

Описується створення інсталяційного пакету для програми. У додаток до інструкції включають рекомендації щодо перенесення:

- програми на інші платформи;

- коду до іншого середовища програмування.

**7. Керівництво користувача**

Керівництво користувача має вклю­чати наступну інформацію.

Загальні відомості:

- позначення і найменування програми;

- клас розв'язуваних задач та (або) призначення програми і відомості про функціональні обмеження її застосування.

Умови застосування програми:

- перелік програмного забезпечення, необхідного для функціонуван­ня програми (бібліотечні файли, драйвери, текстури, тощо);

- тип та характеристики платформи (наприклад, ПК), на якій має працювати програма; тип носія даних та операційної системи;

- вимоги до периферійних пристроїв, плат введення/виведення. Характеристики програми

Наводиться опис основних характеристик і особливостей програми (системні вимоги для коректного запуску, часові характеристики, режим роботи, засоби контролю правильності виконання тощо).

Інсталяція та видалення

Наводиться інструкція по встановленню програми на цільову платфо­рму (наприклад, ПК) та інструкцію її видалення по закінченню роботи. Послідовність дій користування програмою

Описується детальна послідовність дій, а саме:

- підготовка вхідних даних, та налаштунки перед початком роботи;

- опис користувачу послідовності маніпуляцій з діалогом (меню) із відображенням скриншотів інтерфейсу програми;

- опис всіх можливих повідомлень, що видаються користувачу в ході виконання програми; загальна послідовність роботи та опис отриманих ре­зультатів.

**8. Висновки**

Стислий та чітко сформульований виклад основних досягнутих ре­зультатів. Обсяг - не більше однієї сторінки тексту.

**9. Список використаних джерел**

**10. Додатки**

В додатки доцільно включати:

- текст програми (вихідний код програми), якщо він відсутній в ос­новній частині пояснювальної записки;

- результати працездатності програми на конкретних прикладах;

*Текст програми*

Текст програми складається з текстів одного або декількох поймено­ваних розділів, кожний з яких містить текст одного файлу із кодом про­грами.

В текст програми обов'язково потрібно додавати коментарі**,** що ма­ють відображати функціональне призначення й структуру програми.

*Результати роботи програми*

В цьому розділі наводяться приклади функціонування програми під час випробувальних дій та результати перевірки програми на працездат­ність.

**Правила оформлення курсової роботи**

**Загальні вимоги**

Об’єм курсової роботи — 25-30 сторінок (без додатків).

Пояснювальну записку до курсової роботи оформлюють на білому папері формату А4 (210x297), орієнтація — книжна. По­яснювальна записка повинна бути зброшурована і мати обкладинку з текс­том титульного листа.

Текст на сторінці розміщують в один стовпчик. Він має бути середньої щільності та друкується шрифтом чорного кольору Times New Roman, розміром 14.

Міжрядковий інтервал — півтора, з вирівнюванням тексту по ширині рядка та з функцією автоматичного переносу слів. Відступи від країв ар­куша: зверху, знизу і зліва — 20 мм; справа — 10 мм.

Абзацний відступ повинен бути однаковим впродовж усього тексту пояснювальної записки і дорівнювати п'яти знакам.

Текст основної частини роботи розбивають на розділи, підрозділи, пункти та підпункти.

Заголовки структурних частин пояснювальної записки: ЗМІСТ, ВСТУП, ВИСНОВКИ, СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ, ДОДАТКИ не нумерують, інші частини — нумеруються.

Заголовки структурних елементів роботи і заголовки розділів вар­то розташовувати в середині рядка і друкувати прописними літерами без крапки в кінці, не підкреслюючи.

Заголовки підрозділів, пунктів і підпунктів варто починати з абзацно- го відступу і друкувати рядковими літерами, окрім першої.

Крапка в кінці заголовка не ставиться. Якщо заголовок складається з двох або більш речень, вони розділяються крапкою.

Заголовки пунктів друкують рядковими літерами (крім першої пропи­сної) врозрядку, без відступу від тексту. У кінці заголовка, надрукованого без відступу від тексту, ставиться крапка.

Відступ між заголовком (за винятком заголовка пункту) і наступним або попереднім текстом має бути не менше двох рядків. Відстань між основами рядків заголовка, а також між двома заголовками приймають та­ку ж, як у тексті.

Не допускається розміщати найменування підрозділу, пункту, підпун­кту в нижній частині сторінки, якщо після нього розташовано менше 3 ря­дків тексту.

Кожну структурну частину необхідно починати з нової сторінки.

Роздруковані на ПК вихідні коди розробленої програми повинні від­повідати формату основної частини пояснювальної записки. їх включають у загальну нумерацію сторінок і розміщають, як правило, у додатку.

**Нумерація сторінок**

Сторінки пояснювальної записки нумерують арабськими цифрами, дотримуючись наскрізної нумерації по всьому тексту. Номер сторінки про­ставляється у правому верхньому куті сторінки без крапки в кінці.

Титульний лист (перша сторінка), лист завдання (друга сторінка) та анотація не нумеруються, але вони входять до загальної нумерації сторі­нок.

Ілюстрації і таблиці, які розміщуються на окремих сторінках, включа­ють у загальну нумерацію сторінок.

**Нумерація розділів, підрозділів, пунктів, підпунктів**

Розділи, підрозділи, пункти, підпункти нумерують арабськими циф­рами. Розділи повинні мати порядкову нумерацію в межах матеріалу, який викладається, і позначатися арабськими цифрами без крапки. По центру рядка друкується номер розділ без крапки та заголовок розділу.

Підрозділи нумерують у межах кожного розділу. Номер підрозділу складається з номера розділу і порядкового номера підрозділу, розділених крапкою. У кінці номера підрозділу крапка не ставиться, наприклад: 3.2 (другий підрозділ третього розділу). Потім у тому ж рядку йде заголовок підрозділу.

Пункти нумерують у межах кожного підрозділу. Номер пункту скла­дається з порядкового номера розділу, підрозділу, пункту. Між цифрами і у кінці номера ставиться крапка, наприклад: 1.2.4.

Підпункти нумерують у межах кожного пункту по правилам нумерації пунктів.

**Ілюстрації**

Ілюстрації (фотографії, рисунки, схеми, графіки, карти) необхідно розміщувати посередині сторінки, безпосередньо після першого нагаду­вання про них в тексті або на наступній сторінці. На всі ілюстрації мають бути посилання.

Ілюстрація позначається словом Рисунок \_ з додаванням номера. Далі через тире розміщують назву ілюстрації. При необхідності ілюстрації до­повнюють пояснюючими даними (підрисунковий текст).

Ілюстрації варто нумерувати арабськими цифрами по порядку в межах розділу.

Номер ілюстрації складається з номера розділу та порядкового номера ілюстрації, розділених крапкою, наприклад:

Рисунок 2.3 — Схема алгоритму.

Підпис ілюстрації забороняється розміщувати в тексті сторінки за відсутності самої ілюстрації.

Якщо в роботі одна ілюстрація, її нумерують за загальними правила­ми. Якщо ілюстрація не вміщується на одній сторінці, її можна переносити на інші, при цьому назву ілюстрації розміщують на першій, а пояснюючі дані - на кожній сторінці (у відповідності з розміщеними на ній елемента­ми). Ілюстрації на наступних аркушах підписують наступним чином: Рисунок 2.3 — Схема алгоритму, аркуш 2.

Ілюстрації при необхідності можуть бути перераховані в змісті з по­значенням їхніх номерів, найменуванням і номерами сторінок, на яких во­ни розташовані.

**Таблиці**

Цифровий матеріал, як правило, оформлюють у вигляді таблиць.

Горизонтальні та вертикальні лінії, які розмежовують рядки таблиці, а також лінії, що обмежують таблиці з боків, можна не проводити, якщо їхня відсутність не заважає користуванню таблицею.

Таблиця розташовується посередині сторінки, безпосередньо після те­ксту, у якому вона згадується в перший раз або на наступній сторінці. На всі таблиці в тексті мають бути посилання.

Таблиці нумерують арабськими цифрами по порядку в межах поділу.

Номер таблиці складається з номера розділу і порядкового номера таблиці, розділених крапкою, наприклад: Таблиця 2.3 — третя таблиця дру­гого поділу.

Якщо в роботі одна таблиця, її нумерують згідно вищевказаних вимог.

Таблиця може мати назву, яку друкують рядковими літерами (перша прописна) і поміщають над таблицею. Назва повинна бути короткою та ві­дображувати зміст таблиці.

Якщо рядки або графи таблиці виходять за формат сторінки, таблицю варто розділити на частини, розміщуючи одну частину під іншою, або по­руч, або переносячи частину таблиці на наступну сторінку. При цьому в кожній частині таблиці повторюють її голівку і боковик:

При розподілі таблиці на частини допускається її голівку або боковик заміняти відповідно номерами граф або рядків. При цьому графи і/або рядки першої частини таблиці нумерують арабськими цифрами.

Слово «Таблиця » вказують один раз зліва з абзацного відступу над першою частиною таблиці, а над іншими частинами пишуть:

Продовження таблиці \_ із зазначенням її номера.

Заголовки граф таблиці друкують із прописних літер, а підзаголовки - із рядкових, якщо вони складають одне речення з заголовком.

Підзаголовки, що мають самостійне значення, пишуть з прописної лі­тери. У кінці заголовків і підзаголовків таблиць крапки не ставлять. Заго­ловки і підзаголовки граф вказують у єдиному числі.

Таблиці при необхідності можуть бути перераховані в змісті, з зазна­ченням їхніх номерів, назв та номерів сторінок, на яких вони розміщують­ся.

**Переліки**

Переліки при необхідності можуть бути приведені всередині пунктів або підпунктів. Перед переліками ставлять двокрапку.

Перед кожною позицією переліку слід ставити рядкову літеру україн­ського алфавіту зі скобкою або, не нумеруючи, - дефіс (перший рівень де­талізації).

Для подальшої деталізації переліку варто використовувати арабські цифри зі скобкою (другий рівень деталізації).

Переліки першого рівня деталізації друкують рядковими літерами з абзацного відступу, другого — із відступом відносно місця розташування переліків першого рівня.

**Примітки**

Примітки розміщують у тексті при необхідності пояснення змісту тек­сту, таблиці або ілюстрації.

Примітки розміщують безпосередньо після тексту, ілюстрації т.д., до яких вони відносяться. Одна примітка не нумерується.

Слово «Примітка» друкують з прописної літери, з абзацного відступу, після слова «Примітка» ставлять крапку і з прописної літери в тому ж ряд­ку подають текст примітки.

Декілька приміток нумерують послідовно арабськими цифрами з кра­пкою. Після слова "Примітки" ставлять двокрапку і з нового рядка з абзац­ного відступу після номера примітки з прописної літери подають її текст.

**Формули та рівняння**

Формули і рівняння розташовують безпосередньо після їхнього нага­дування в тексті, посередині сторінки.

Вище і нижче формули (рівняння) треба залишити не менше одного вільного рядка.

Формули і рівняння нумеруються порядковою нумерацією в межах розділу.

Нумерують тільки ті формули та рівняння, на які є посилання в тексті.

Номер формули або рівняння включає в себе номер розділу та поряд­ковий номер формули (рівняння), розділених крапкою, наприклад: (2.3) — третя формула другого розділу.

Номер формули (рівняння) вказують у тому ж рядку у крайньому пра­вому положенні.

Пояснення значень символів і числових коефіцієнтів варто розміщува­ти безпосередньо під формулою і у тій же послідовності, у якій вони вико­ристовуються у формулі.

Пояснення значення кожного символу і числового коефіцієнта варто давати з нового рядка. Перший рядок пояснення починають з абзацу сло­вом «де» без двокрапки.

Переносити формули чи рівняння на наступний рядок допускається тільки на знаках виконуваних операцій, повторюючи знак операції на по­чатку наступного рядка. Коли переносять формули чи рівняння на знакові операції множення, застосовують знак «х».

Формули, що йдуть одна за одною й не розділені текстом, відокрем­люють комою.

**Посилання**

Посилання в тексті пояснювальної записки на джерела слід зазначати порядковим номером за переліком посилань, виділеним двома квадратни­ми дужками, наприклад, «...у роботах [1-7] ...».

При посиланнях: на розділи, підрозділи, пункти, ілюстрації, таблиці, формули, рівняння, додатки, зазначають їх номери.

При посиланнях в тексті пояснювальної записки слід писати: «... у розділі 4 ...», «... дивись 2.1 ...», «... за 3.3.4 ...», «... відповідно до 2.3.4.1 ...», або «... на рисунку 1.3 ...», «... у таблиці 3.2 ...», «... за формулою (4.1) ...», «... у рівняннях (1.23) - (1.25)...», «... у додатку Б ...».

**Список використаних джерел**

Приклади оформлення бібліографічних записів:

Книга одного автора:

Меженный O.A. Turbo Pascal. Самоучитель. [Текст] / O.A. Меженный. — М. : Издательский дом «Вильяме», 2005. — 336 с.

Книга двох авторів:

Либерти Д. Освой самостоятельно С++ за 21 день [Текст] : 5-е изда­ние [перевод с английского В.А. Коваленко] / Д. Либерти, Б. Джонс. — М. : Издательский дом «Вильяме», 2006. — 784 с.

Книга під редакцією:

Сухарев M.B. Основы Delphi. Профессиональный подход. [Текст] : учеб. пособие / под ред. М.В. Финкова. — СПб.: Наука и Техника, 2004. — 600 с.

Стаття із періодичного видання (журналу, газети):

Денищенко Н. Resource Governor: управление ресурсами в SQL Server 2008 [Текст] / H. Денищенко // RSDN Magazine. — 2007.№ 4. — С. 65—71.

Електронний ресурс:

Программирование в Delphi глазами хакера. [Электронный ресурс]. — СПб. : БХВ — Петербург, 2001. — 1 электрон, опт. диск (CD-ROM).

Мережевий ресурс:

Шредер Карла. Установка и сопровождение программного обеспече­ния в системах на базе Debian [Электронный ресурс] / Карла Шрёдер. — Издательство «Питер», 2006. — Режим доступу: <http://www.rsdn.ru/article/linux/debian.xml> (18.03.2009).

**Додатки**

Додатки варто оформлювати як продовження пояснювальної записки на наступних сторінках, або, якщо це доцільно, окремим документом. Роз­міщують додатки в порядку їхньої появи в тексті.

Додатки нумерують літерами українського алфавіту за винятком літер Ґ, Є, З, І, ї, О, Ч, Ь. Якщо додаток один, то він позначається як «А».

Якщо додатки оформлюють на сторінках після основного тексту запи­ски, то кожен з них повинен починатися з нової сторінки.

Додаток оформлюється наступним способом: всередині рядка вгорі сторінки друкується слово ДОДАТОК \_ з вказівкою його порядку (літери), далі з нового рядка друкується заголовок рядковими літерами з першої прописної і розміщується по центру щодо тексту сторінки.

Якщо додатки оформлюються окремим документом, на титульному аркуші під назвою теми курсової роботи прописними літерами друкують слово ДОДАТКИ.

При необхідності текст додатка може мати розділи, підрозділи, пункти і підпункти, які варто нумерувати в межах кожного додатка, наприклад: А. 5 - п'ятий розділ додатка А.

Загальним вимогам нумерації підкоряються і таблиці (Таблиця Б.З - третя таблиця додатка Б), рисунки, формули і т.д.

Джерела, що цитуються тільки в додатку, розглядають незалежно від основної частини і повинні бути перераховані в кінці кожного додатка в переліку використаних джерел.

Форма цитування, правила упорядкування переліку використаних джерел, і виноски повинні бути аналогічними приведеним в основній час­тині. Перед номером цитати і відповіднім номером в переліку і виносках ставлять позначення додатка.

**Теми курсових робіт**

Тема обирається за номером в списку групи. Для першої групи починаючи з варіанту №1, для другої — з варіанту №31.

|  |  |
| --- | --- |
| Варіант | Тема |
| 1 | **Визначення приналежності точки фігурі.** Розробити і закодувати алгоритм, що дозволяє розмістити по вказів­ках користувача N фігур із заданого переліку (круг, кільце, квадрат, ромб, трикутник, сектор) у визначених користувачем розмірі, місці, орієнтації і комбінації (в т.ч. з накладанням) і визначити чи належить задана координатами X, У точка одній (або декільком) з цих фігур. |
| 2 | **Математичні операції над матрицями.** Розробити і закодувати алгоритм, що дозволяє введення матриці да­них, редагування і запам'ятовування її, а також виконувати визначені в матричній алгебрі операції над матрицями (транспонування, скла­дання, віднімання, множення на скаляр, скалярне і векторне множен­ня, обернення, визначення детермінантів і так далі). |
| 3 | **Моделювання елементарних алгебраїчних функцій.** Змоделювати поліноміальну функцію, раціональну, статичну, пока­зову, логарифмічну. Результати вивести на екран у вигляді графіків. |
| 4 | **Операції зі звичайними та десятковими дробами.** |
| 5 | **Моделювання руху більярдних шарів.** Розробити і закодувати алгоритм, що дозволяє вважаючи екран біль­ярдним столом, зобразити рух 3-х куль (червоного, синього і жовто­го) відповідно до законів геометричної оптики (тобто кут віддзерка­лення дорівнює куту падіння), змінювати випадковим чином швид­кість кулі після віддзеркалення, та при зіткненні куль імітувати спа­лах, наприклад, білим кольором, і міняти забарвлення куль випадко­вим чином або на допоміжний колір до кольорів куль, що приймають участь у зіткненні, або для простоти імпульс руху куль, що зіткнули­ся, може залишатися незмінним. |
| 6 | **Пошук найкоротших шляхів.** Розробити і закодувати алгоритм пошуку шляху від фіксованої вер­шини до всіх вершин (алгоритм Форда-Беллмана, алгоритм Данцига) та алгоритм пошуку усіх найкоротших шляхів (пошук шляхів у безконтурному графі). |
| 7 | **Моделювання тригонометричних, зворотних тригонометричних і гіперболічних функцій.** Змоделювати тригонометричні, зворотні тригонометричні і гіпербо­лічні функції. Результати вивести на екран у вигляді графіків. |
| 8 | **Операції з римським цифрами.** |
| 9 | **Моделювання хвиль при падінні каменя на поверхню води.** Розробити і закодувати алгоритм, що дозволяє імітувати круги на во­ді, вважаючи, що «камінь» падає в центр екрану, і кожна нова хвиля має свій колір, що задається: а) природним спектром, б) випадково, в) іншим способом. Задавати «каменям» різну вагу, що випадково змінюється, наприклад, рівномірно, відповідно до чого: а) змінювати ширину кілець, або б) змінювати швидкість кілець, або в) давати су­марний колір кілець, що наклалися в цьому місці. Частота падіння «каменів» чергується: або постійна, або випадкова. |
| 10 | **Побудова матриці зв'язків по матриці з'єднань довільного графа.** Розробити і закодувати алгоритм, що дозволяє побудувати матрицю зв'язків по матриці з'єднань довільного графа. |
| 11 | **Розв'язання систем нелінійних рівнянь.** Розробити програму, що дозволяє розв’язувати системи нелінійних рівнянь, введені користувачем. |
| 12 | **Моделювання погоні вовка за зайцем.** Розробити і закодувати алгоритм, що дозволяє визначити поведінку вовка, який хоче наздогнати зайця, якщо початкові координати і швидкості учасників задаються користувачем перед початком роботи алгоритму і: а) заєць рухається по прямій, б) заєць рухається по ана­літичній кривій, в) у зайця «ступор» і в кожну наступну одиницю ча­су він рухається у випадковому напрямку, г) інша траєкторія. |
| 13 | **Організація лінійного списку.** Скласти програму, яка містить поточну інформацію про книги в біб­ліотеці. Відомості про книги містять: номер УДК; Прізвище і ініціа­ли автора; Назва; Рік видання; Кількість екземплярів в бібліотеці. Програма повинна забезпечити:  - початкове формування даних у вигляді лінійного списку;  - додавання даних про книги, що поступають в бібліотеку;  - зміна даних при введенні інформації про те, що користувач бере або повертає книгу;  видачу даних про наявність книг в бібліотеці. |
| 14 | **Розв'язання систем лінійних рівнянь.** Розробити програму, що дозволяє розв’язувати системи лінійних рівнянь, введені користувачем. |
| 15 | **Задача про Робін Гуда.** Розробити і закодувати алгоритм, що дозволяє користувачеві визна­чити з якою швидкістю Робін Гуду потрібно кинути камінь через огорожу в кріпосній стіні, щоб той потрапив у віконце темниці, якщо зріст Робін Гуда, розміри і висота огорожі, вікна і відстань між всіма елементами задаються користувачем перед початком роботи алгори­тму. Розв'язання повинно бути анімаційним. |
| 16 | **Шифрування текстових файлів.** Розробити і закодувати алгоритм, що дозволяє шифрувати і розшиф­ровувати текст трьома методами: зміщення, зміщення на N символів, зміщення на випадкове N число символів, або іншим методом. |
| 17 | **Розв'язання алгебраїчних рівнянь із дійсними та комплексними коефіцієнтами.** Розробити програму, що дозволяє розв’язувати алгебраїчні рівняння з дійсними та комплексними коефіцієнтами, введені користувачем. |
| 18 | **Моделювання польоту тривимірного об'єкта.** Розробити і закодувати алгоритм, що моделює «обліт» тривимірного твердого тіла по заданій траєкторії (наприклад, тверде тіло - тетра­едр, траєкторія - еліпс). |
| 19 | **Розпізнавання правильних скобкових послідовностей для обчислення арифметичних виразів.** Перетворення інфіксного виразу у постфікс - ний вираз (зворотній польський запис ЗПЗ). Обчислення постфіксного виразу (виразу по його ЗПЗ). |
| 20 | **Побудова буфера глибини.** Розробити і закодувати алгоритм буфера глибини для зображення сцени, що складається із випадково розташованих в просторі багатокутників. |
| 21 | **Растрове представлення геометричних об'єктів**. Розробити і закодувати алгоритм, що дозволяє відображати растрове представлення геометричних об'єктів (алгоритм Брезенхема, викрес­лювання лінії, кола, еліпса по точках). |
| 22 | **Моделювання спіралей.** Розробити та закодувати алгоритм, що дозволяє будувати двомірні спіралі (Архімедова спіраль, спіраль Ферма, гіперболічна, логарифмічна). |
| 23 | **Побудова зображень Платонових тіл.** Розробити і закодувати алгоритм побудови дротяного (каркасного) зображення фігур Платонових тіл (тетраедр, куб, октаедр, ікосаедр, додекаедр). |
| 24 | **Моделювання циклоїд.** Розробити та закодувати алгоритм, що дозволяє будувати циклоїдальні криві. |
| 25 | **Обчислення визначених інтегралів.** Розробити програму, що дозволяє обчислювати визначенні інтеграли, введені користувачем. |
| 26 | **Розрахунок остового дерева.** Розробити і закодувати алгоритм, що дозволяє розрахувати остове дерево і виведення його на екран (із забезпеченням масштабування і скролінгу) для автоматичної класифікації об'єктів, заданих їх описа­ми (таблицею «об'єкт - властивості») або матрицею зв'язків (алго­ритм «Краб»). |
| 27 | **Розпізнавання ланцюгів символів.** Розпізнавання належності слова певній регулярній мові за допомо­гою алгоритмів кінцевих автоматів (алгоритм Кнута-Моріса-Прата). |
| 28 | **Побудова бінарного дерева.** Автоматизована інформаційна система на залізничному вокзалі міс­тить відомості про відправлення поїздів дальнього слідування. Для кожного поїзда вказується: номер поїзда, станція призначення, час відправлення. Скласти програму, яка:   1. забезпечує початкове введення даних в інформаційну систему і формування дерева; 2. проводить виведення всього дерева; 3. вводить номер поїзда і виводить дані про цей поїзд;   вводить назву станції призначення і виводить дані про всі поїзди, які слідують до цієї станції. |
| 29 | **Гра «Сапер»** |
| 30 | **Побудова остових дерев.** Породження всіх остовів дерева (графа). Побудова остового дерева мінімальної ваги за методом Крускала. Побудова остового дерева мінімальної ваги за методом Прима. |
| 31 | **Обчислення спеціальних функцій.** Розробити і закодувати алгоритм для обчислення наступних функцій: гіпергеометрична функція, сферична, циліндрова, функція Ейрі. |
| 32 | **Гра «Судоку».** Програма повинна мати демонстраційний режим та режим гри. |
| 33 | **Операції із двадцятип'ятизначними числами** |
| 34 | **Розрахунок ієрархічного дерева.** Розробити і закодувати алгоритм , що дозволяє розраховувати ієрархічне дерево для автоматичної класифікації в евклідовій метриці об’єктів, заданих їх описами (таблицею «об’єкт — властивості») або матрицею зв’язків і вивести на екран дендрограму із забезпеченням її масштабування та скролінгу. |
| 35 | **Створення лабіринту.** Розробити та закодувати алгоритм, що дозволяє створити прямокутний лабіринт, який має один вхід та один вихід, тобто має одне рішення. |
| 36 | **Інтерполяція та екстраполяція.** |
| 37 | **Спектральній аналіз на основі швидкого перетворення Фур'є.** Розробити і закодувати алгоритм швидкого аналізу на основі швидкого перетворення Фур'є. |
| 38 | **Обчислення значення похідної в точці.** |
| 39 | **Імітація спектральних складових світла.** Розробити і закодувати алгоритм , що дозволяє розмістити на екрані N точок, заданих їх координатами X,Y (вони можуть задаватись або генеруватись випадково). Алгоритм повинен забарвлювати точки , змінюючи колір згідно природного спектру від центру екрану до його периферії: а) випадково-рівномірно; б) випадково нормально; з’єднати а) кожну точку з кожною; б) точки з різною парністю; в) точки з однаковою парністю – змінювати забарвлення ліній та швидкість випадковим чином. |
| 40 | **Броунівський рух молекул.** Розробити та закодувати алгоритм, що моделює броунівський рух молекул. |
| 41 | **Спектральний аналіз на основі дискретного перетворення Фур'є.** Розробити і закодувати алгоритм спектрального аналізу на основі дискретного перетворення Фур'є. |
| 42 | **Побудова фракталів.** Розробити та закодувати алгоритм, що дозволяє будувати наступні фрактали:крива Мінковського, трикутник Серпінського, множина Мандельброта. Надати можливість змінювати параметри цих фракталів. |
| 43 | **Синтаксичний аналізатор.** Реалізація алгоритму синтаксичного аналізу тексту. Побудова дере­ва. Пірамідальне сортування, оцінка його ефективності. |
| 44 | **Операції з числами в вісімковій системі.** Розробити калькулятор, що дозволяє здійснювати операції з числами в вісімковій системі, а також переводити з/в неї в десяткову. |
| 45 | **Гра «Змійка»** |
| 46 | **Моделювання кривих 3-го порядку** |
| 47 | **Лексичний аналізатор.** Реалізація алгоритму лексичного аналізу тексту. |
| 48 | **Розв’язання нелінійних та трансцендентних рівнянь** |
| 49 | **Розрахунок та візуалізація дії сили Архімеда** |
| 50 | **Побудова фракталів.** Розробити та закодувати алгоритм, що дозволяє будувати наступні фрактали: сніжинка Коха, Н-фрактал, папороть Барнслі. Надати можливість змінювати параметри цих фракталів. |
| 51 | **Розрахунок опору електричних ланцюгів** |
| 52 | **Універсальний календар літочислення** |
| 53 | **Розрахунок та візуалізація спектру зв’язків.** Розробити і закодувати алгоритм , що дозволяє розраховувати і візуалізувати спектр зв’язків об’єктів, що класифікуються, заданих їх описами (таблицею «об’єкт - властивості») і матрицею зв’язків. Алгоритм повинен забезпечувати масштабування та скролінг спектру (алгоритм «Спектр») |
| 54 | **Операції з числами в шістнадцятковій системі.** Розробити калькулятор, що дозволяє здійснювати операції з числами в шістнадцятковій системі, а також переводити з/в неї в десяткову. |
| 55 | **Гра «Морський бій»** |
| 56 | **Перевірка належності точки багатокутнику.** Розробити та закодувати алгоритм, що визначає, де знаходиться задана точка відносно заданого багатокутника: всередині, на границі або поза межами. |
| 57 | **Моделювання кривих 4-го порядку** |
| 58 | **Гра «Тетріс»** |