

Лабораторна робота 1

Варіант 2с

Реалізувати «навчальну базу даних» – систему для роботи з даними певного типу. Необхідно реалізувати підтримку наступних операцій:

1. Додавання елементів;
2. Зберігання даних (запис даних у файл);
3. Відновлення даних (зчитування даних з файлу);
4. Вивід всіх збережених даних;
5. Пошук за заданими критеріями (див. підваріанти а-с);
6. Модифікація елементів (додаткові бали);
7. Видалення елементів (додаткові бали).

Для зберігання елементів треба реалізувати наступні механізми:

1. Зберігання в пам'яті, без збереження на диск (можна використати довільну структуру даних, зокрема бібліотечні структури на зразок `std::vector`);
2. Зберігання у вигляді текстового файлу;
3. Зберігання у вигляді бінарного файлу

Для кожного елементу, окрім описаних у відповідному варіанті даних, треба також зберігати унікальний ID – ціле число, яке буде унікальним для кожного елементу даних.

Необхідно реалізувати зберігання даних, які вводить користувач, а також генерацію випадкових даних для зберігання та пошуку. У випадку зберігання даних на диск, ці дані мають залишатись доступними після повного закриття та перезапуску програми. Тобто користувач може повністю закрити програму, і навіть перезавантажити ОС, а після нового запуску програми всі раніше збережені дані залишаються доступними і програма може їх прочитати.

Необхідно реалізувати наступні режими роботи для реалізованої програми:

1. Інтерактивний діалоговий режим – коли користувач може вибирати, які операції виконувати та задавати необхідні параметри.
2. Демонстраційний режим – задаються фіксовані значення параметрів та послідовності виконання операцій, що демонструють правильність роботи операцій в різних ситуаціях. В цьому режимі користувач не має нічого вводити.
3. Автоматичний режим "benchmark" з вимірами ефективності роботи різних механізмів зберігання.

В автоматичному режимі "benchmark" задається певне число елементів N; генерується N випадкових елементів, вони додаються та зберігаються, потім відновлюються і виконується пошук. При цьому вимірюється час виконання операцій (з точністю до мілісекунд) та

розмір збережених даних (в байтах). Бажано окремо виміряти час виконання кожної з операцій. Ця процедура виконується для всіх реалізованих механізмів зберігання. Потім значення N збільшується, і виміри повторюються для нового значення N. Така процедура повторюється, доки час виконання операцій не стане більшим за 10 секунд. Результати вимірів для всіх розглянутих значень N треба зберегти і надати викладачу під час здачі лабораторної (наприклад, додавши відповідні файли в репозиторій з кодом виконаних завдань).

Варіант 2

Інформація про товари в магазині. Зберігаються наступні дані про кожен товар: назва товару; одиниці виміру – кілограми, літри, поштучно, пакунки; кількість товару (у відповідних одиницях, дійсне число); дата та час виробництва; термін зберігання (кількість діб з дати виробництва, не більше 10 років). Критерії пошуку:

Пункт с

Товари з назвою, що завершується на заданий фрагмент тексту; товари із заданою одиницею вимірювання та терміном зберігання не менше заданого; товари з датою виробництва у заданому діапазоні.

Лабораторна робота 2

Варіант 2

Підваріант 2

В цій лабораторній необхідно реалізувати задану структуру даних, використовуючи три різні підходи:

1. На основі масиву фіксованого розміру. При виході за межі масиву видавати повідомлення про помилку.
2. На основі масиву змінного розміру (ArrayList, наприклад, std::vector).
3. На основі зв'язного списку. Самостійно обрати однозв'язний чи двозв'язний список та інші деталі реалізації.

Необхідно реалізувати наступні режими роботи для реалізованої програми:

1. Інтерактивний діалоговий режим – коли користувач може вибирати, які операції виконувати та задавати необхідні параметри.
2. Демонстраційний режим – задаються фіксовані значення параметрів та послідовності виконання операцій, що демонструють правильність роботи структур даних та операцій в різних ситуаціях. В цьому режимі користувач не має нічого вводити.
3. Автоматичний режим "benchmark" з вимірами ефективності роботи різних версій структур даних.

В автоматичному режимі необхідно виміряти та порівняти час виконання кожної операції окремо для всіх трьох підходів реалізації – на основі масиву фіксованого розміру, масиву змінного розміру та зв'язних списків. Можна додатково виміряти розмір пам'яті, який займають структури даних (і отримати додаткові бали). Варто виміряти час виконання для різних розмірів структур даних та різної кількості операцій. Результати вимірів для всіх розглянутих розмірів, операцій та сценаріїв треба зберегти та надати викладачу під час здачі лабораторної.

Всі версії структур даних, режими роботи, операції та сценарії мають бути реалізовані в рамках однієї програми (тобто один виконуваний файл, одна функція main(), довільна кількість файлів з кодом).

Варіант 2

Реалізувати структуру даних «стек» з наступними операціями: 1) create_empty – створення пустого стеку; 2) push – додавання елемента до стеку; 3) pop – вилучення елемента зі стеку; 4) peek – значення верхнього елемента стеку (без видалення); 5) is_empty – перевірка на пустоту.

Підваріант 2

В структурі зберігаються дати (день, місяць, рік)

