

Лабораторна робота № 23

Тема. Обробка файлів з структурованими даними.
Мета. Формування вмінь і навиків створення та використання файлів з структурованими даними. Закріплення вмінь і навиків використання функцій вводу-виводу та обробки рядків. Застосування вмінь і навиків програмування алгоритмів лінійної, розгалуженої та циклічної структури.

Контрольні запитання.

1. Коли та з якою метою доцільно використовувати файли з структурованими даними в програмуванні?
2. У чому переваги та недоліки файла з структурованими даними стосовно файлів послідовного доступу?
3. Що відбувається на рівні операційної системи при відкритті файла?
4. Для чого визначається структура запису?
5. Як підрахувати кількість байтів поля-рядка у структурованому файлі?
6. Як визначити зміщення у структурованому файлі для зчитування значення вказаного поля визначеного запису?
7. Як закрити файл? Що відбувається на рівні операційної системи при закритті файла?

Завдання.

1. **Скласти дві програми: першу – для створення структурованого двійкового файла довільного доступу, який зберігає дані списку по варіанту, другу - для опрацювання даних, записаних у цей файл.**

Вимоги до програми:

- значення даних першої програми ввести з клавіатури з відповідною перевіркою коректності типів та даних;
- перед відкриттям файла для обробки в другій програмі передбачити перевірку існування файла. При відсутності файла видати відповідне повідомлення та завершити виконання програми;
- результати виконання другої програми вивести в одному діалоговому вікні.

Варіанти:

1. Список студентів містить їх прізвища, імена та по батькові. Вияснити, чи є в даному не впорядкованому за алфавітом списку студентів задане прізвище. Якщо є, то вказати порядкові номери таких прізвищ, відповідні імена та по батькові;
2. Список студентів містить їх прізвища, імена та по батькові. Вияснити, чи є в даному не впорядкованому за алфавітом списку студентів однофамільці. При виявленні однофамільців надрукувати їх дані;
3. Список студентів містить їх прізвища, імена, по батькові та абрєвіатури груп. Визначити, в яких групах зустрічається прізвище *Поліщук* (списки груп не впорядковані за алфавітом);
4. Заданий список, що містить перелік прізвищ студентів і назви вулиць, на яких вони проживають. Визначити, на якій з вулиць пропиває найбільша кількість студентів;
5. Список студентів містить їх прізвища, імена та по батькові. Визначити, яке ім'я зустрічається найчастіше;
6. Список містить прізвища студентів і назви улюблених предметів кожного з них. Визначити, який з предметів вказаний найбільшу кількість разів і для скількох студентів він є улюбленим;
7. Список містить прізвища студентів та суми отриманих балів кожним з них. Впорядкувати список за спаданням суми отриманих балів;
8. Список містить перелік футбольних команд та кількість очок, набраних у чемпіонаті кожною з них. Вказати назву команди-чемпіона. При наявності кількох команд з максимальною кількістю набраних очок надрукувати назви всіх таких команд;
9. Список співробітників містить їх прізвища, імена, по батькові і номери їх домашніх телефонів. Знайти телефон співробітника за прізвищем або вказати на відсутність даних у списку. Якщо для співробітника зазначено декілька телефонів, то вивести їх всі;
10. Заданий список, що містить перелік прізвищ студентів і назви вулиць, на яких вони проживають. Вказати, чи є серед студентів такі, що проживають по вулиці *Соборній*;
11. Заданий список, що містить перелік прізвищ студентів і назви вулиць, на яких вони проживають. Вказати кількість і прізвища студентів, що проживають по вулиці *Дубенській*.
12. Список містить відомості про автомобілі: їх марки, номери і прізвища власників. Вказати кількість автомобілів даної марки;
13. Список містить відомості про автомобілі: їх марки, номери і прізвища власників. Вказати прізвища власників і номери автомобілів даної марки;
14. Список містяться відомості про кожну книгу: прізвище автора, рік та назву видання, Знайти назви книг даного автора;

15. Список містить дані про кожну партію експортованих товарів: назву, країну, що імпортує товар та об'єм партії, що поставляється в штуках. Вказати загальний об'єм експорту даного товару.
16. Список містить дані про книги з програмування (прізвище автора і його ініціали, назва книги, рік видання). Вивести на екран дані останньої виданої книги;

Наприклад, програма для формування файлу завдання варіанту № 16 може мати вигляд:

```
Book[] masBook = new Book[] { new Book { name = "Изучаем С#", author = "Єндрю Стіллен, Джениффер Грин" },
                              new Book("Кобзар", "Тарас Григорович Шевченко", 1840),
                              new Book(true), new Book(false), new Book(), book,book1,book2};

const string plusStr = "
int lenZap = 206;
FileStream f = new FileStream("I:\\LR20.dat", FileMode.Create);
BinaryWriter w = new BinaryWriter(f, Encoding.Unicode);
int i;
for (i = 0; i < masBook.Count(); i++)
{w.Write((masBook[i].author+plusStr).Substring(0,50));
 Console.WriteLine(w.BaseStream.Position);
 w.Write((masBook[i].name+plusStr).Substring(0,50));
 Console.WriteLine(w.BaseStream.Position);
 w.Write(masBook[i].year);
 Console.WriteLine(w.BaseStream.Position);
}
w.Close();
```

А програма для розв'язання цього завдання може бути такою:

```
f = new FileStream("I:\\LR20.dat", FileMode.OpenOrCreate);
BinaryReader r = new BinaryReader(f,Encoding.Unicode);
int countZap=(int)r.BaseStream.Length/lenZap;
if (countZap == 0)
{Console.WriteLine("Дані книг відсутні");
 Console.ReadLine();
}
r.BaseStream.Seek(202,SeekOrigin.Begin); //позиціонуємо на початок в базовому потоці
int index = 0, year, lastYear = r.ReadInt32();
for (i = 1; i < countZap; i++)
{r.BaseStream.Seek(i*lenZap+202,SeekOrigin.Begin); //позиціонуємо на початок в базовому потоці
 year = r.ReadInt32();
 if (year > lastYear)
 {lastYear = year;
 index = i;
}
}
//зчитуємо дані найсвіжішої книги
r.BaseStream.Seek(index*lenZap,SeekOrigin.Begin); //позиціонуємо на початок в базовому потоці
book.author=r.ReadString().TrimEnd();
book.name = r.ReadString().TrimEnd();
book.year = r.ReadInt32();
r.Close();
Console.Write("Дані найсвіжішої книги: ");
book.Info();
Console.ReadLine();
```

2. Скласти програму для допису даних у структурований двійковий файл довільного доступу, створений в результаті виконання першої програми попереднього завдання. Після допису даних у файл знову виконати програму обробки його даних (тобто другу програму попереднього завдання). Порівняйте розміри файлів, створених програмами попередньої та цієї лабораторної роботи.