МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЛЬВІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ІВАНА ФРАНКА

Факультет електроніки і комп’ютерних технологій

**Звіт**

Про виконання лабораторної роботи № 1

## «**Знов структури. Файли.**»

**Виконав:**

Студент групи ФЕП-14с

Грицюк Максим Олегович

**Перевірив:**

ас. Левуш П. Н.

Львів 2025

**Мета:** Повторити матеріал, засвоєний у курсі Програмування мовою С. Ознайомитись із інтерфейсом користувача оболонки Microsoft Visual Studio.

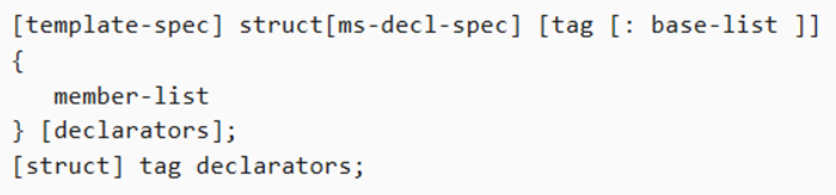
**Обладнання:**

Комп'ютер з встановленим програмним забезпеченням(Microsoft Visual Studio/Clion).

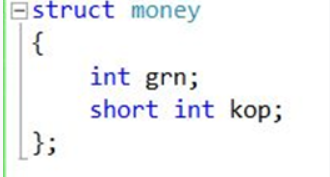
**Теоретичні відомості**

**3.1. Структура у мові С/С++**

Для оголошення структур використовується ключове слово struct. Синтаксис оголошення є наступним:



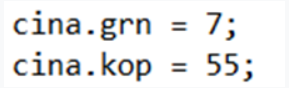
Приклад такого оголошення структури:



Створення змінної типу нашої структури буде виглядати наступним чином

C:\Users\Home-PC\Documents\ShareX\Screenshots\2025-02\chrome_sXxcsPzdUG.png

Для присвоєння значень полям структури використовується крапка у вигляді імя\_змінної.назва\_поля і буде виглядати для нашого прикладу наступним чином



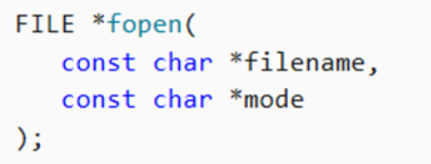
Або можна виконати ці дві операції в одну стрічку:

C:\Users\Home-PC\Documents\ShareX\Screenshots\2025-02\chrome_YCFWtzxlgz.png

Далі доступ до полів структури можна також отримати через крапку.

**3.2. Робота з текстовими файлами у С/С++ (у стилі мови С)**

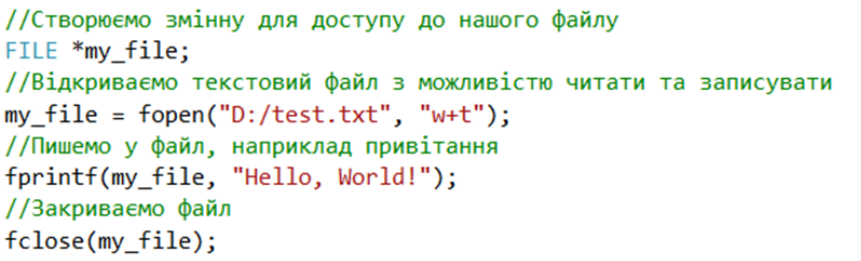
Для доступу до текстового файлу у мові С передбачену наступну конструкцію:



Яка описана у бібліотеці

C:\Users\Home-PC\Documents\ShareX\Screenshots\2025-02\chrome_FtiPI33dCI.png

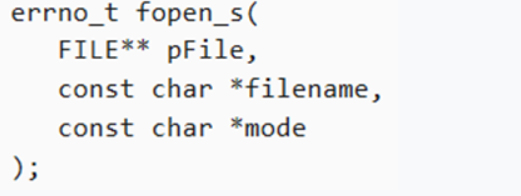
Як це робиться можна продемонструвати наступним прикладом



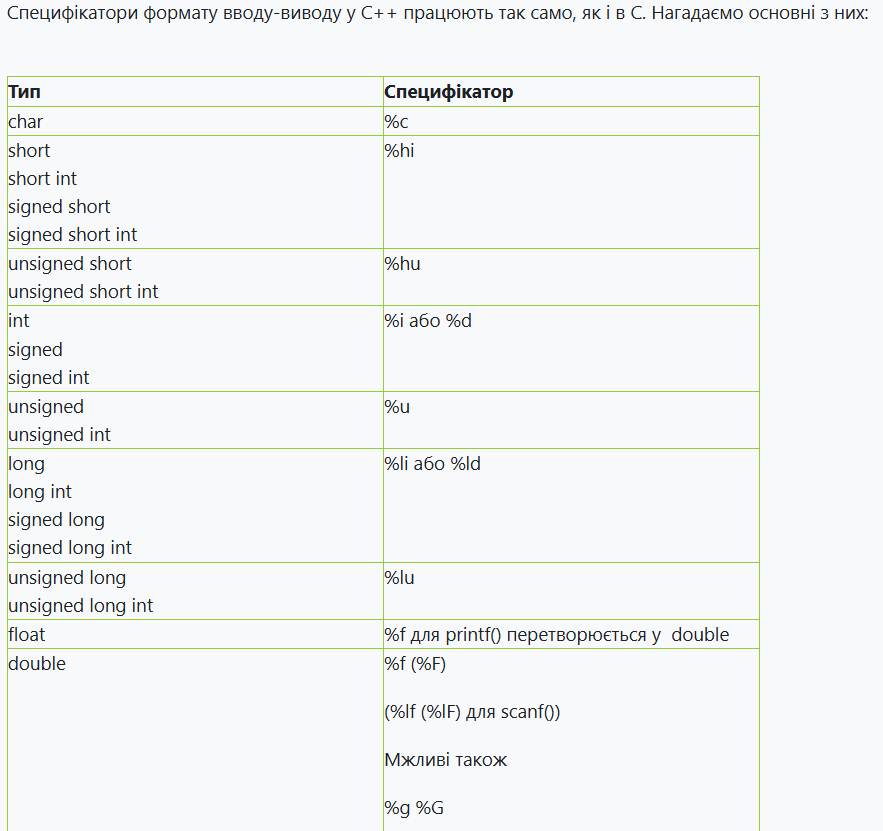
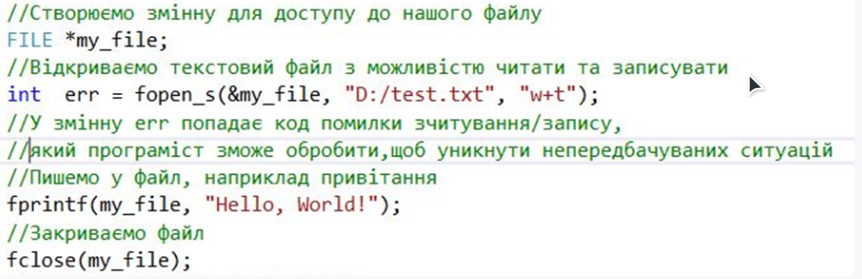
Однак, такий метод доступу вважається застарілим і при спробі компіляції проєкту може виникнути помилка С4996

C:\Users\Home-PC\Documents\ShareX\Screenshots\2025-02\chrome_dqrTS9k3Wl.png

Тому доцільно використати більш сучасну конструкцію



Тоді наш приклад доступу до текстового файлу буде виглядати наступним чином:



**Хід роботи**

**Завдання:** Розв’язати наступну задачу, написавши виконуваний код мовою С++ у середовищі MS VS. Якщо з об’єктивних причин немає можливості використовувати MS VS, можна застосувати будь-яке інше середовище (тоді п.2 пропускається), інші завдання залишаються.

Як усім відомо, у нашій країні виведені з обігу монети до 10 коп. Але ціни залишились у старому форматі. У зв’язку з цим виникає така задача:

Кожна ціна складається з двох чисел, гривні та копійки. Для такої структури визначити операції додавання та множення на додатне число. Також створити метод, який буде заокруглювати кількість копійок за правилами Нацбанку.

Вхідними параметрами програми будуть ціни на товари та їх кількість, наприклад:

Пакет молока 900г, 19 грн 89 коп, 3 шт;

Батон Нарізний, 13 грн 29 коп, 1 шт

**На виході має бути загальна сума чеку  та сума до оплати із заокругленням.**

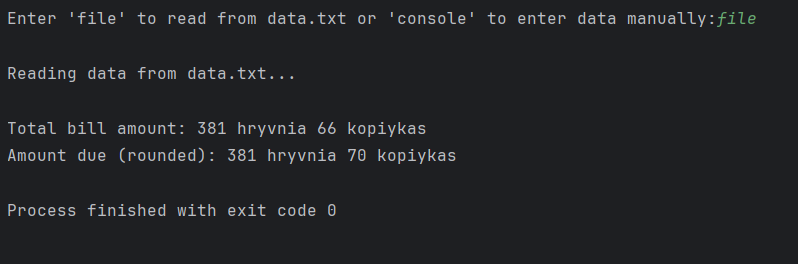
**Файл price.h:** містить структуру Price, яка зберігає вартість у гривнях і копійках. Також тут оголошені методи. Тут знаходяться лише оголошення, а вся їхня реалізація винесена в price.cpp.

**Файл price.cpp:** тут знаходиться реалізація всіх методів і функцій, оголошених у price.h. Конструктори створюють об'єкти Price, метод normalize() стежить за правильним збереженням копійок, а оператори + і \* виконують відповідні математичні операції. Функція roundToNationalBank() відповідає за округлення ціни.

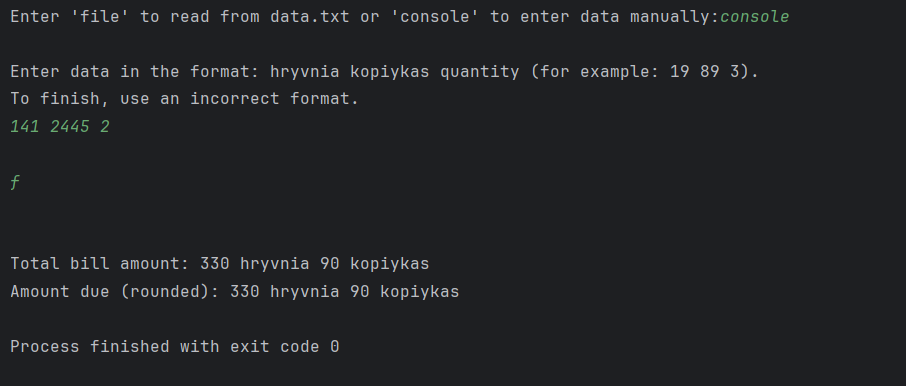
**Файл ConsoleApplication1.cpp:** головний файл програми, який взаємодіє з користувачем. Програма в циклі очікує введення ціни товару та його кількості, у випадку якщо користувач не захотів використовувати дані по замовчуванню, або ж файл просто не вдалося відкрити, розраховує загальну вартість і додає її до підсумку. Коли користувач вводить некоректні дані, програма завершує підрахунок і виводить суму, округлену згідно з правилами національного банку. Всі обчислення виконуються за допомогою функцій із price.h, а головний файл забезпечує введення та виведення.

**Демонстрація:**

**Випадок №1.** Ми обирємо зчитування даних з текстового файлу.

****

**Випадок №2.** Ми власноруч вписуємо необхідні дані.



**Висновок:** у цій роботі було розділено програму на три файли: заголовковий файл price.h, файл реалізації price.cpp та головний файл ConsoleApplication1.cpp. Такий підхід зробив код більш структурованим і зручним для роботи. Логіка обробки цін була винесена в окремий модуль, що дозволяє легко змінювати чи допрацьовувати її без впливу на основний код. Головний файл тепер відповідає лише за взаємодію з користувачем. У результаті програма коректно підраховує загальну вартість товарів і округлює суму за правилами Національного банку.

**p.s** Посилання на github: https://github.com/unknownpanic/OOP\_lab1