

Виконав: Сенченко Віталій Вікторович;  
Група: комп'ютерна математика 1

Умова task 1 :

### **Робота з файлами на Сі та Сі++. Особливості роботи з бінарними та текстовими файлами.**

Робота з файлами в С і С++ зазвичай виконується за допомогою функцій стандартної бібліотеки С. Наприклад, `fopen`, `fread`, `fwrite` та `fclose` є найпоширенішими функціями, які використовуються для відкриття, читання, запису та закриття файлів відповідно.

Під час роботи з бінарними файлами дані зчитуються та записуються безпосередньо у вигляді байтів, без будь-якого форматування чи кодування. Це забезпечує швидшу обробку даних, оскільки файл не потрібно перетворювати з текстового на двійковий. Однак працювати з двійковими файлами також складніше через відсутність структури та труднощі з налагодженням.

При роботі з двійковими файлами важливо розуміти особливості зберігання та доступу до даних. Двійкові файли записуються як потік байтів і зберігаються як послідовність байтів. Дані, що зберігаються у двійковому файлі, повинні читатися в тому ж порядку, в якому вони були записані. Як результат, програміст повинен знати формат даних і порядок байтів під час роботи з бінарними файлами.

Під час роботи з текстовими файлами дані зчитуються та записуються як текст, який форматується для читання людиною. Це полегшує роботу з ними та налагодження, але також уповільнює обробку, оскільки дані потрібно перетворювати з текстового на двійковий і назад.

Незалежно від того, є файл двійковим чи текстовим, важливо не забувати закривати файл після його використання, щоб звільнити ресурси та запобігти пошкодженню даних.

- **Відкриття та закриття файлів**

У С і С++ ви можете використовувати функцію `fopen()`, щоб відкрити файл, і функцію `fclose()`, щоб закрити файл.

Ось приклад того, як відкрити файл для читання:

```
// Open a file for reading
FILE* file = fopen("file.txt", "r");

// Check if the file was successfully opened
if (file == NULL) {
    // Handle the error
}

// Here you use the file

// Close the file
fclose(file);

return 0;
```

Перший аргумент `fopen()` — це ім'я файлу, який потрібно відкрити, а другий аргумент — режим, у якому відкривається файл. У цьому прикладі режим «r», що означає «читати».

Функція `fopen()` повертає тип `FILE*`, який є вказівником на об'єкт `FILE`. Якщо файл було успішно відкрито, `FILE*` вказуватиме на об'єкт `FILE`. Якщо файл не вдалося відкрити, `fopen()` поверне `NULL`.

Важливо завжди перевіряти значення, яке повертає `fopen()`, і, якщо необхідно, обробляти помилку, як показано в прикладі вище.

Щоб закрити файл, ви можете скористатися функцією `fclose()`, яка приймає `FILE*` як аргумент. Це закриває файл і звільнить усі ресурси, які використовував об'єкт `FILE`.

- **Текстові файли**

Текстові файли — це файли, які містять текст, який читає людина. У C та C++ ви можете використовувати функції `fprintf()` та `fscanf()` для запису та читання з текстових файлів відповідно.

Ось приклад того, як записати текстовий файл на C++:

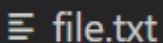
```
// Open a file for writing
FILE* file = fopen("file.txt", "w");

// Write some text to the file
fprintf(file, "some text\n");
fprintf(file, "some text 2\n");

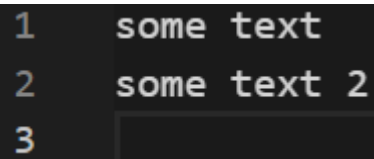
// Close the file
fclose(file);

return 0;
```

Якщо заглянути в директорію, де розміщений виконуваний код, то побачимо:



Вміст файлу:



Функція `fprintf()` працює подібно до функції `printf()`, за винятком того, що вона приймає `FILE*` як свій перший аргумент, а вихід записується у файл, а не на консоль.

Для читання з текстового файлу можна скористатися функцією `fscanf()`, яка працює подібно до функції `scanf()`. Ось приклад того, як читати з текстового файлу:

```
// Open a file for reading
FILE* file = fopen("file.txt", "r");

// Read a line of text from the file
char line[50];
fscanf(file, "%[^\n]", line);

// Print the line of text
printf("%s\n", line);

// Close the file
fclose(file);

return 0;
```

Запустивши цей код, отримаємо в консолі:

```
some text
```

- **<fstream> бібліотека**

Працювати з файлами на C і C++ можна за допомогою бібліотеки `fstream`. Ця бібліотека забезпечує інтерфейс, який дозволяє програмам отримувати доступ до файлів на жорсткому диску. Бібліотеку `fstream` можна використовувати для відкриття, читання та запису як текстових, так і двійкових файлів.

Під час роботи з текстовими файлами бібліотека `fstream` надає функції для читання та запису рядків символів. Ця бібліотека також надає функції для читання та запису двійкових файлів, які є файлами, що складаються з необроблених даних, які можна використовувати для більш складних операцій.

Бібліотека `fstream` також надає функції для маніпулювання та пошуку у файлі. Наприклад, бібліотека надає функції для встановлення поточної позиції у файлі та для пошуку відносно поточної позиції. Це значно полегшує роботу з великими файлами.

Таким чином, бібліотека `fstream` надає інтерфейс для роботи як з текстовими, так і з бінарними файлами на C і C++. Бібліотека також надає функції пошуку всередині файлу, що значно полегшує роботу з великими файлами.

Приклад використання бібліотеки:

```
std::ifstream inputFile("file.txt");

if (inputFile.is_open()) {
    // File was successfully opened
    std::cout << "File was opened successfully." << std::endl;
} else {
    // File could not be opened
    std::cout << "Error opening file." << std::endl;
}
```