C -code for Video Transmission

This readme provides a detailed explanation of the file transfer program, its functionalities, usage instructions, and technical aspects.

Project Overview

This program offers a user-friendly graphical user interface (GUI) for transferring files between two designated folders on your system. It caters specifically to video files but can be modified to handle other file types as well.

Key Features:

- **Multi-file Selection:** Select and transfer multiple video files simultaneously, enhancing efficiency.
- **Concurrent Transfer:** Utilizes threads to ensure smooth transfer of multiple files at once, optimizing transfer time.
- **End-of-File Handling:** Reliably handles end-of-file (EOF) markers to guarantee complete file transfer.
- **User-friendly Interface:** Provides a clear and intuitive GUI for selecting files, monitoring progress, and receiving transfer completion notifications.
- **Error Reporting:** Includes error handling mechanisms to address potential issues during file transfer and report them in a pop-up message.

Prerequisites

- **GTK+ Libraries:** Ensure you have GTK+ libraries installed on your system. These libraries provide the graphical user interface functionalities used by the program. You can usually find installation instructions for your specific operating system online.

Building and Running the Program

1. **Save the Code:** Save the provided C code as a file with a .c extension (e.g., file_transfer.c).
2. **Compile the Code:** Compile the code using a C compiler with GTK+ libraries linked. Here's an example command using gcc:

```
gcc file_transfer.c -o file_transfer `pkg-config --cflags --libs gtk+-3.0`
```

  - Replace pkg-config --cflags --libs gtk+-3.0 with the appropriate command for your system to link the GTK+ libraries during compilation. You can find these commands in your GTK+ documentation.

3. **Run the Program:** Open a terminal window in the directory where you saved the compiled program (file_transfer). Then, execute the program using the following command:

```
./file_transfer
```

User Guide

1. **Launching the Program:** Upon running the program, a window titled "File Transfer Program" will appear.
2. **Selecting the Source Folder:**
   - You'll see a heading and a label prompting you to choose the source folder containing your video files.
   - Two buttons labeled "Folder1" and "Folder2" represent the folders.
   - Click the button corresponding to the folder where your videos are located.
3. **Displaying Folder Contents:**
   - Clicking a folder button will display the list of files within that folder in the main window.
   - Each file (ideally a video) will be listed with a checkbox next to it.
4. **Selecting Files for Transfer:**
   - Select the checkboxes for the video files you want to transfer to the destination folder.
5. **Initiating Transfer:**
   - A button labeled "Add Selected Files" will appear below the list of files once you've selected some files.
   - Click "Add Selected Files" to begin the transfer process.
6. **Description Button:**
   - An additional button labeled "Description" is provided in the main window.
   - Clicking this button opens a separate window displaying a detailed explanation of the project, including its functionality and implementation details.
7. **Transfer Progress and Results:**
   - The program operates silently in the background while transferring the selected files.
   - Once the transfer is complete, a pop-up message will appear, indicating success or failure.

    - If successful, the message will confirm that the files were transferred successfully.

    - In case of failure, the message will inform you about any errors encountered during the transfer process.

Technical Overview

- **Multithreading:** The program utilizes threads to enable concurrent transfer of multiple files, significantly improving transfer speeds compared to a single-threaded approach.

- **Error Handling:** The code incorporates error handling mechanisms to address potential issues such as file access errors, invalid file paths, or errors during data transfer. These errors will be reported in the pop-up message upon transfer completion.

- **File I/O:** The program utilizes file I/O operations to read from the source files, segment them into smaller chunks for efficient transfer, and write them to the destination folder.

- **End-of-File (EOF) Handling:** The program includes logic to properly detect the end of a file being transferred and signal the completion of the transfer process for that specific file.

Additional Notes

- This program is designed primarily for video files. However, with modifications to file handling logic, it could be adapted to handle other file types as well.