**DAY- 2**

**Objective: Develop a console-based file explorer application in C++ that interfaces with the Linux operating system to manage files and directories.**

**PSEUDOCODE**

Here is the pseudocode for the file explorer application:

Initialize Application

Display welcome message

While True

Display available operations

Get user input for desired operation

If user selects "Move file"

Get source and destination file paths

Move file from source to destination

Else If user selects "Copy file"

Get source and destination file paths

Copy file from source to destination

Else If user selects "Delete file"

Get file path to delete

Delete file

Else If user selects "Create file"

Get file path to create

Create file

Else If user selects "Search file"

Get search criteria and directory

Search for files matching criteria in directory

Display search results

Else If user selects "Manage file permissions"

Get file or directory name and new permissions

Modify file or directory permissions

Display permission modification results

Else

Display invalid operation message

Log operation details

Check operation status

If operation was successful

Continue normal execution

Else

Handle error and display error message

If user selects "Exit"

Break out of loop and exit application

Exit Application

Display goodbye message

**Explanation:**

**Initialize Application**

This section sets up the application and displays a welcome message to the user.

**While True**

This loop continues to run until the user chooses to exit the application.

**Display available operations**

This section displays a list of available operations to the user, such as "Move file", "Copy file", etc.

**Get user input for desired operation**

This section gets the user's input for the desired operation.

**If-Else statements**

These statements check the user's input and perform the corresponding operation.

For example, if the user selects "Move file", the application will get the source and destination file paths and move the file.

**Log operation details**

This section logs the details of the operation, such as the type of operation and any relevant file paths.

**Check operation status**

This section checks whether the operation was successful or not.

If the operation was successful, the application continues normal execution.

If the operation failed, the application handles the error and displays an error message.

**Exit Application**

This section displays a goodbye message and exits the application when the user chooses to quit.

Overall, this pseudocode outlines the basic structure and logic of a file explorer application. It includes user input, operation execution, error handling, and logging.