Data Structures and Algorithms

BS (CIS) – PIEAS

Semester / Term Project Report

Title: File Deduplication System

Group Members:

- **Hira Arif**
- **Laiba Batool**
- **Mariyam Muzammil**

Instructor: Dr. Naveed Akhtar
Course Code: CIS-318
Submission Date: December 15, 2024

---

1. Project Overview

The File Deduplication System is a software application designed to identify duplicate files within a specified directory. It utilizes efficient algorithms and data structures to process and compare file contents, enabling users to eliminate redundant files, save storage space, and maintain an organized file system. The system provides an interactive interface for users to review duplicates and decide whether to delete them.

`

2. Program Functionality and User Interaction

Functionality:

- Scans a specified directory to identify all files.
- Uses cryptographic hashing (SHA-256) to calculate unique file signatures.
- Organizes files into hash-based buckets for efficient duplicate detection.
- Provides options to view, verify, and delete duplicate files.

User Interaction:

- Users specify the directory to be scanned.
- The program processes files and identifies duplicates.
- Detected duplicates are displayed in a user-friendly format.
- Users are prompted to confirm the deletion of duplicate files, which are then moved to the Recycle Bin.

New to Us:

- Implementing hashing techniques for file comparison.
- Designing and working with a custom linked list and hash map.
- Integrating Windows-specific API functions to move files to the Recycle Bin.

---

3. Time Breakdown

Phase	Time Spent
Investigation & Research	10 hours
Design & Algorithm Development	15 hours
Coding	1 week
Testing & Debugging	3 days
Documentation	5 hours
Total Time:	10 days and 5 hours

---

4. Algorithms Used

SHA-256 Hashing Algorithm:

Pseudo-code:

```
Initialize hashing context using OpenSSL.
While reading file in chunks:
    Update hash context with the chunk data.
Finalize hash computation.
Convert the result to a hexadecimal string and return.
```

File Comparison Algorithm:

Pseudo-code:

```
Compute hashes for both files:
    Call SHA-256 Hashing Algorithm for each file.
If hashes do not match:
    Return false (files are not identical).
If hashes match:
    Open both files and compare byte by byte.
Return true if contents are identical; otherwise, return false.
```

Hash Function for Buckets:

Pseudo-code:

```
Input: File hash as a string.
Compute a numeric hash using std::hash.
Map the result to a bucket index by calculating hash % numBuckets.
Return the bucket index.
```

---

5. Data Structures Used

Linked List:

- Stores file entries in buckets for duplicate detection.
- Class: **List**
    - Nodes: Each node contains a FileEntry object and a pointer to the next node.

Hash Map:

- Organizes files into buckets based on their hash values.
- Class: **HashMap**
    - Buckets are implemented as linked lists to handle collisions.

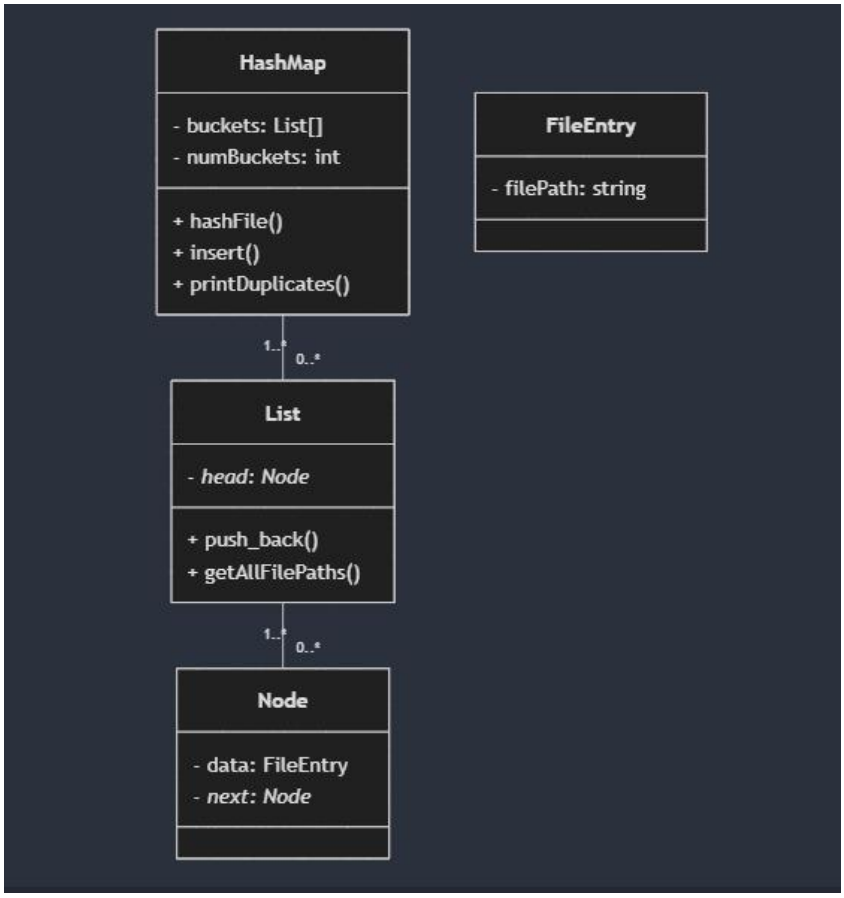
FileEntry Structure:

- Holds metadata about each file, such as its path.

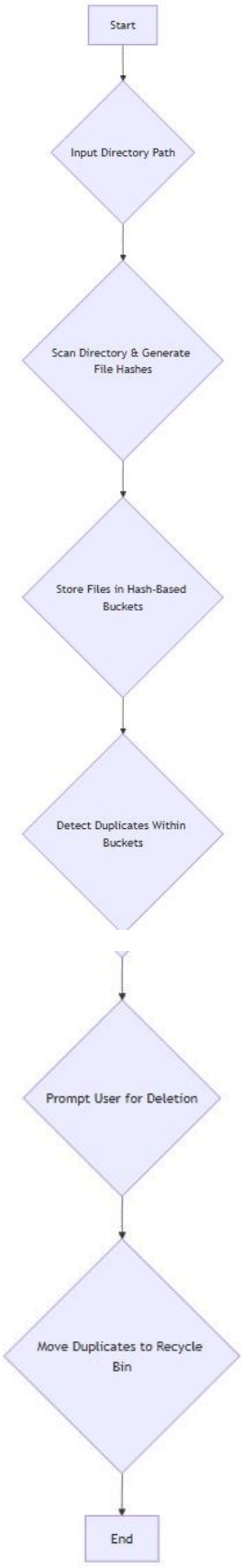
---

6. Class Diagrams and Flowcharts

Class Diagram:



Flowchart:


Start
Input Directory Path
Scan Directory & Generate File Hashes
Store Files in Hash-Based Buckets
Detect Duplicates Within Buckets
Prompt User for Deletion
Move Duplicates to Recycle Bin
End

7. Group Member Contributions

Equal contribution of each member to everything

---

8. Challenges Faced

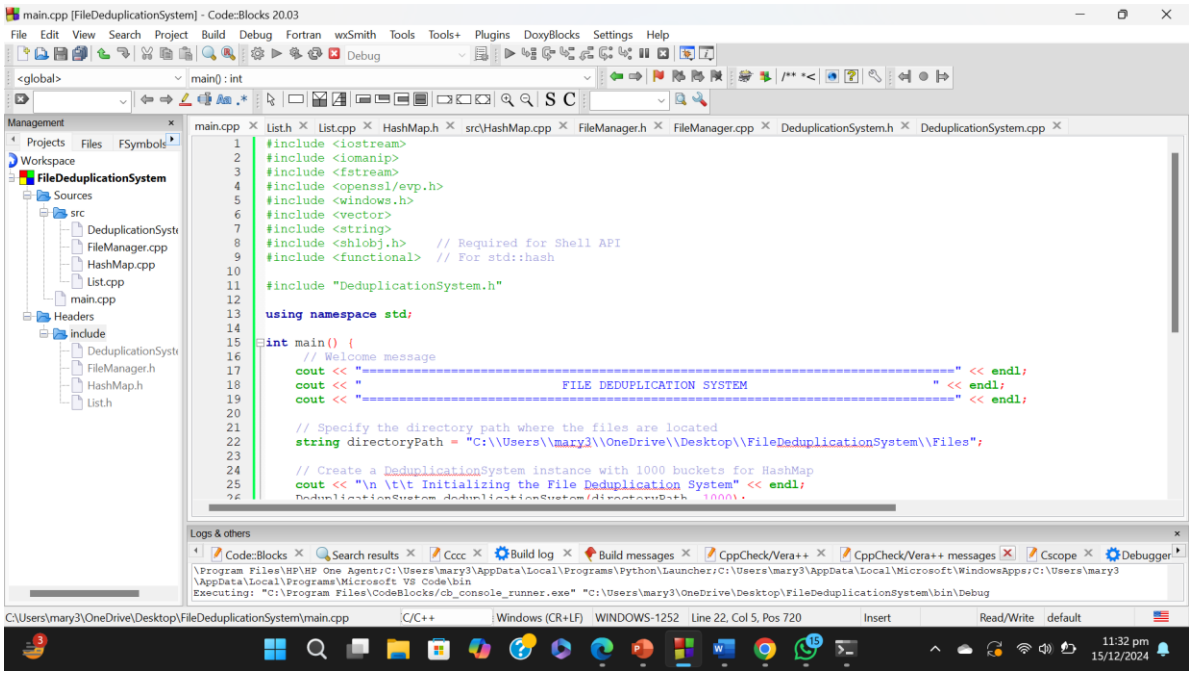
- Understanding and using OpenSSL for hashing.
- Efficiently handling file operations and memory management.
- Ensuring cross-platform compatibility.

---

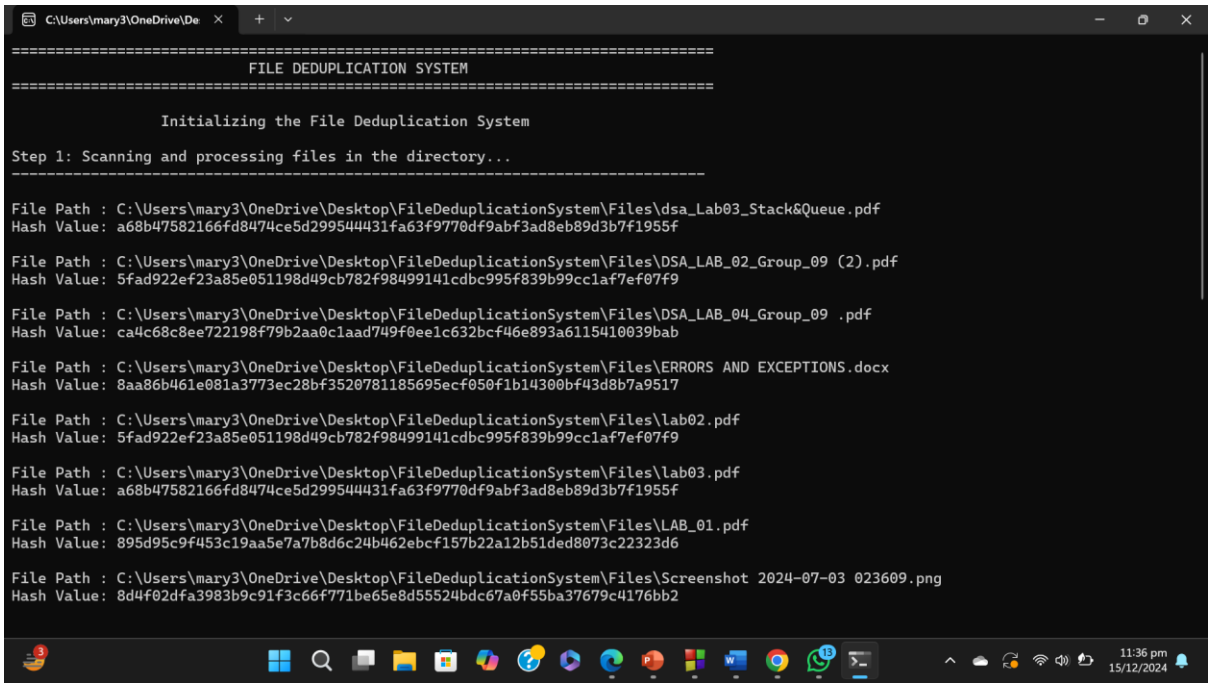
9. Conclusion

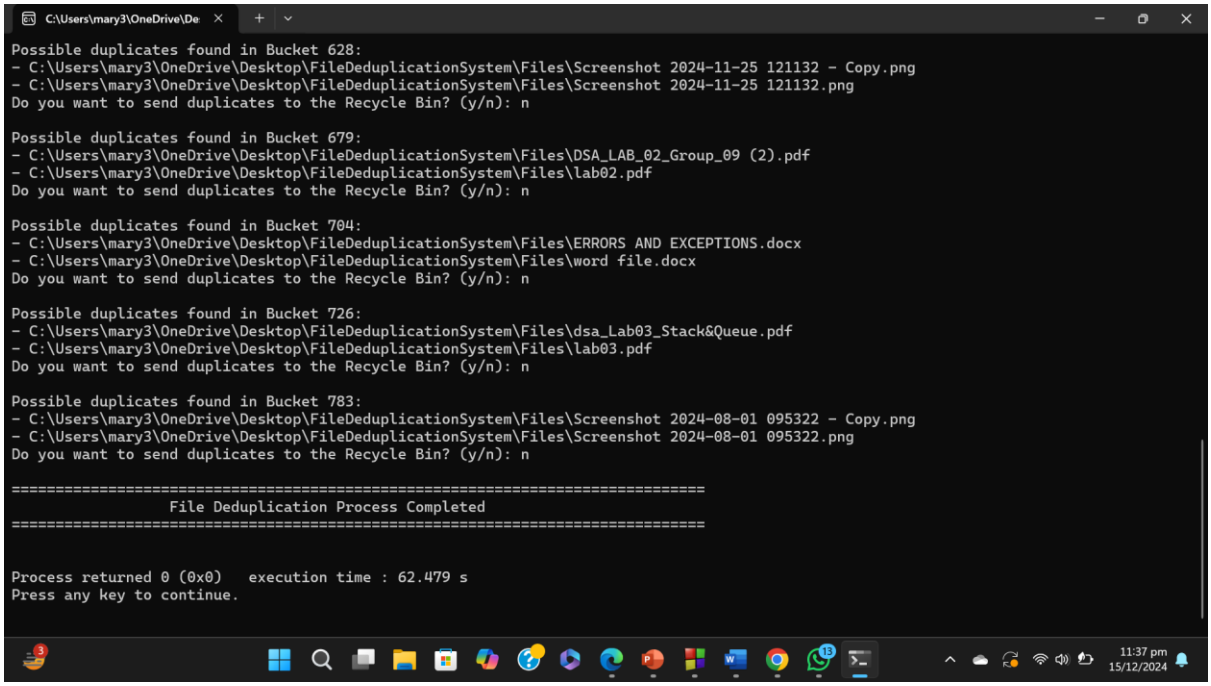
The File Deduplication System successfully identifies and handles duplicate files. By leveraging efficient data structures like linked lists and hash maps, it ensures optimal performance even with large file sets. The project provided hands-on experience with cryptographic hashing, advanced C++ programming, and Windows API usage. It has practical applications for file management and storage optimization.

10.Output and Directory Structure


main.cpp [FileDeduplicationSystem] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
Management
Projects Files FSymbols
Workspace
FileDeduplicationSystem
Sources
src
DeduplicationSyste
FileManager.cpp
HashMap.cpp
List.cpp
main.cpp
Headers
include
DeduplicationSyste
FileManager.h
HashMap.h
List.h
main.cpp List.h List.cpp HashMap.h src\HashMap.cpp FileManager.h FileManager.cpp DeduplicationSystem.h DeduplicationSystem.cpp
#include <iostream>
#include <iomanip>
#include <fstream>
#include <openssl/evp.h>
#include <windows.h>
#include <vector>
#include <string>
#include <shlobj.h> // Required for Shell API
#include <functional> // For std::hash
#include "DeduplicationSystem.h"
using namespace std;
int main() {
// Welcome message
cout << "==" << endl;
cout << " FILE DEDUPLICATION SYSTEM " << endl;
cout << "==" << endl;
// Specify the directory path where the files are located
string directoryPath = "C:\\Users\\mary3\\OneDrive\\Desktop\\FileDeduplicationSystem\\Files";
// Create a DeduplicationSystem instance with 1000 buckets for HashMap
cout << "\n \t\t Initializing the File Deduplication System" << endl;
Logs & others
Code::Blocks Search results Cccc Build log Build messages CppCheck/Vera++ CppCheck/Vera++ messages Cscope Debugger
\Program Files\HP\HP One Agent;C:\Users\mary3\AppData\Local\Programs\Python\Launcher;C:\Users\mary3\AppData\Local\Microsoft\WindowsApps;C:\Users\mary3\AppData\Local\Programs\Microsoft VS Code\bin
Executing: "C:\Program Files\CodeBlocks/cb_console_runner.exe" "C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\bin\Debug
C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\main.cpp C/C++ Windows (CR+LF) WINDOWS-1252 Line 22, Col 5, Pos 720 Insert Read/Write default
11:32 pm
15/12/2024

Output:


C:\Users\mary3\OneDrive\De
FILE DEDUPLICATION SYSTEM
Initializing the File Deduplication System
Step 1: Scanning and processing files in the directory...
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\dsa_Lab03_Stack&Queue.pdf
Hash Value: a68b47582166fd8474ce5d299544431fa63f9770df9abf3ad8eb89d3b7f1955f
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\DSA_LAB_02_Group_09 (2).pdf
Hash Value: 5fad922ef23a85e051198d49cb782f98499141cdbc995f839b99cc1af7ef07f9
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\DSA_LAB_04_Group_09 .pdf
Hash Value: ca4c68c8ee722198f79b2aa0c1aad749f0ee1c632bcf46e893a6115410039bab
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\ERRORS AND EXCEPTIONS.docx
Hash Value: 8aa86b461e081a3773ec28bf3520781185695ecf050f1b14300bf43d8b7a9517
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\lab02.pdf
Hash Value: 5fad922ef23a85e051198d49cb782f98499141cdbc995f839b99cc1af7ef07f9
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\lab03.pdf
Hash Value: a68b47582166fd8474ce5d299544431fa63f9770df9abf3ad8eb89d3b7f1955f
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\LAB_01.pdf
Hash Value: 895d95c9f453c19aa5e7a7b8d6c24b462ebcf157b22a12b51ded8073c22323d6
File Path : C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\Screenshot 2024-07-03 023609.png
Hash Value: 8d4f02dfa3983b9c91f3c66f771be65e8d55524bdc67a0f55ba37679c4176bb2
11:36 pm
15/12/2024


C:\Users\mary3\OneDrive\De
Possible duplicates found in Bucket 628:
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\Screenshot 2024-11-25 121132 - Copy.png
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\Screenshot 2024-11-25 121132.png
Do you want to send duplicates to the Recycle Bin? (y/n): n
Possible duplicates found in Bucket 679:
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\DSA_LAB_02_Group_09 (2).pdf
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\lab02.pdf
Do you want to send duplicates to the Recycle Bin? (y/n): n
Possible duplicates found in Bucket 704:
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\ERRORS AND EXCEPTIONS.docx
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\word file.docx
Do you want to send duplicates to the Recycle Bin? (y/n): n
Possible duplicates found in Bucket 726:
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\dsa_Lab03_Stack&Queue.pdf
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\lab03.pdf
Do you want to send duplicates to the Recycle Bin? (y/n): n
Possible duplicates found in Bucket 783:
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\Screenshot 2024-08-01 095322 - Copy.png
- C:\Users\mary3\OneDrive\Desktop\FileDeduplicationSystem\Files\Screenshot 2024-08-01 095322.png
Do you want to send duplicates to the Recycle Bin? (y/n): n
File Deduplication Process Completed
Process returned 0 (0x0) execution time : 62.479 s
Press any key to continue.
11:37 pm
15/12/2024