CSE314 Bash Scripting Online - Section B2

May 18, 2025

File Deduplicator
Time: 40 minutes

Overview

As a new member of the IT team at DataSense Analytics, you've been assigned to optimize storage usage across the company's servers. The storage management team has identified duplicate files as a significant waste of disk space, especially in the research department where multiple analysts often save copies of the same datasets. Your task is to create a bash script that will identify and manage duplicate files across directories while keeping a proper record of actions taken.

This script will become a standard tool in the company's storage management toolkit. Successfully completing this task will showcase your scripting skills and attention to detail, potentially fast-tracking your path to more advanced system administration responsibilities.

Objectives

1. Scan one or more directories recursively to identify duplicate files by first initial checking by size (files having different size cannot be duplicates) and then comparing MD5 checksums. This is very important that you first filter by size as computing checksum can be computationally heavy task. However, compute MD5 checksum using the following command –
   `md5sum <file_path> | cut -d ‘ ’ -f 1`

2. Remove duplicate files, keeping one copy

3. Generate a comprehensive report having
   - Total number of files in the beginning
   - Number of duplicates found
   - Space saved by deletion

Sample Directory Structure

- 📁 **research_data/** (Before deduplication)
  - 📁 project1/
    - 📄 dataset_2025.csv 8.2 MB
    - 📄 results.pdf 2.4 MB
    - 📄 analysis.txt 340 KB
  - 📁 project2/
    - 📄 backup_dataset.csv 8.2 MB (duplicate of dataset_2025.csv)
    - 📄 summary.txt 120 KB
  - 📁 project3/
    - 📄 report.pdf 2.4 MB (duplicate of results.pdf)
    - 📄 notes.txt 340 KB (duplicate of analysis.txt)
    - 📄 data_copy.csv 8.2 MB (duplicate of dataset_2025.csv)

- 📁 **research_data/** (After deduplication)
  - 📁 project1/
    - 📄 dataset_2025.csv 8.2 MB (original kept)
    - 📄 results.pdf 2.4 MB (original kept)
    - 📄 analysis.txt 340 KB (original kept)
  - 📁 project2/
    - 📄 summary.txt 120 KB
  - 📁 project3/

Report.txt

```
Total files scanned: 8
Duplicates found: 4
Space saved: 19.14 MB (you may write in plane bytes)
```

Command

```
./deduplicator.sh <target_directory>
```

Detailed Requirements

- The script must accept a single top-level directory as an argument and recursively scan all its subdirectories for files.

- Use file size as an initial filter: files of different sizes are not considered duplicates. This is very important that you first filter by size as computing checksum can be computationally heavy task.
- The script must remove duplicate files, keeping a single original file. Keeping any one of them should suffice.

Hint

- Use associative arrays or dictionaries to group files by size and under the same size where number of elements is more than one, use another dictionary for grouping by MD5 hash.
- Use `find` to get the files recursively.

Submission Guidelines

Important Notes

- Please note that any usage of the internet is strictly prohibited during the assignment.
- Use of any unfair means will be duly punished.
- Submit the source code in Moodle. Failure to submit in Moodle will result in a mark deduction.
- The filename should be `online-StudentID.sh` (e.g., `online-2105XXX.sh`).
- Make sure your script is executable.