Synchronize Photo Collections by Date

Background:

You and two friends each have a common photo collection organized by date, with each day's photos kept in separate folders identified by the date (e.g., "2024-04-15"). Each photo within a day's folder can be uniquely identified by a hash value. It's possible that none of you have the complete "database" for each day. Your task is to create a program that helps all three of you compile a complete collection for each specific day, identifying which photos are missing from your collection and which friend has them.

Requirements:

1. Photo Identification:
   - Assume each photo can be uniquely identified by a hash value (e.g., SHA-256). Simulate this with unique identifiers for each photo within each day's collection. Do not work with photos directly. Handling only the ids is sufficient for the scope of this task.
2. Incremental Synchronization Algorithm:
   - Develop a method to identify differences between the photo collections for each day.
   - Calculate a hash for each day's collection of photo identifiers to quickly determine if synchronization is needed.
   - If hashes differ, identify which specific photo identifiers are missing from your collection and which friend has them.
3. Efficiency Considerations:
   - Use hash comparisons to minimize the computational overhead by avoiding full set operations on days where collections are already identical.
4. Output:
   - For each day, return a list of identifiers missing from your collection, along with a reference to which friend's collection they can be found in.
5. Test Cases:
   - Include scenarios with different overlaps between collections for specific days, testing both the hash comparison and the identification of missing photos.
6. GitHub Repository:
   - Place your code in a GitHub repository, including any test files. Make sure the repository is well-organized, with a clear README file that describes the purpose of the program and how to run it.

Deliverables:

- A GitHub repository containing:
    - A script/program in the programming language of your choice that implements the day-by-day synchronization algorithm, including preliminary hash comparison and detailed identification of missing photos.
    - A set of test inputs and corresponding outputs demonstrating the functionality of your program.