

When I create the file I put in the snapshot message as “Start”.

This program is a simple file version management system implemented in C++. It allows creating, modifying, snapshotting, rolling back, and tracking multiple versions of files. Additionally, it provides system-wide analytics like listing the most recently modified files and files with the largest version trees.

The program uses:

- Tree structures (TreeNode) to maintain file version history.
- Hash maps (VersionMap and Filemap) for fast lookup of versions and files.
- Heaps (RecentHeap and BiggestTrees) for analytics on recently modified and largest files.

Features

1. File Operations:

- CREATE <filename> — Create a new file.
- READ <filename> — Read the content of the active version.
- INSERT <filename> <content> — Append content to the active version.
- UPDATE <filename> <content> — Overwrite content of the active version.
- SNAPSHOT <filename> <message> — Create a snapshot with a message.
- ROLLBACK <filename> — Revert to the parent version.
- ROLLBACK <filename> <version_id> — Revert to a specific version ID.
- HISTORY <filename> — Display the snapshot history of the file.

2. System-Wide Analytics:

- RECENT_FILES <n> — Display the n most recently modified files.
- BIGGEST_TREES <n> — Display the n files with the largest version trees.

In CREATE function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename already exists then it will output “A file with this filename already exists.”. If some content is given along with the filename and CREATE then it will be ignored.

In READ function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”.

In INSERT function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”. If content input is empty then it will output "Error: No content given.". In all other cases, it will insert the content in the file and output the content it has inserted and in which file.

In UPDATE function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”. If content input is empty then it will output "Error: No content given.". In all other cases, it will update the content in the file and output the content it has updated and in which file.

In SNAPSHOT function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”. If message input is empty then it will output "Error: No message given.". If the active version is already a snapshot then it will output "It is already a snapshot.". In all other cases, it will output the file which was snapshotted with the message.

In ROLLBACK function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”. If the version id doesn’t exist in the file then it will output "Error: Version ID given is not valid". If no id is given and the active version is already at the root then it will output "Parent doesnt exist. ". In all other cases, if id is given it will rollback to that id and if no id is given then it will rollback to the parent of the active version.

In HISTORY function – If there is no filename given it will output “Error: No filename given.” If the file corresponding to the filename doesn’t exist then it will output “No such file exists.”. It will ignore the content even if it is or it is not there.

In RECENT_FILES function – If number input is negative then it will output "Error: Number input needs to be positive.". If number input is greater than the size of the heap then it will output till the size of the heap then stop. In all other cases, it will output the filenames in descending order of their last modification times. There is no tiebreaker since modification times will always be different.

In BIGGEST_TREES function – If number input is negative then it will output "Error: Number input needs to be positive.". If number input is greater than the size of the heap then it will output till the size of the heap then stop. In all other cases, it will output the filenames in descending order of their total number of versions. There will be a tiebreaker. I will put the file with a modification time that is more recent above (in the sense that it will be printed first in the order) the file with a modification time that is older (the one with lesser time would be printed first).

If the command given is not one of these then it will output the "Error: The command is invalid.".