VERSION CONTROL USING PERSISTENT BST

Group Members

Rakesh Pavan 17IT154
Yogesh Choubey 17IT252
Dhruvik Navadiya 17IT225
Neeraj Deshpande 17IT226

OVERVIEW:

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later.

GitHub is a popular open source version control system . People use GitHub on a daily basis, to store code with all the revision history they need.

In our project, we have implemented Git like structure using Persistent Binary Search Trees.

It supports adding, deleting, viewing and editing files.

New branch can be created anytime, preserving the previous state in such a way that it can be accessed and modified when needed.

Persistent Data Structures

A Persistent data structure is a data structure that allows preserving the previous version of itself when it is modified.

A data structure is Partially persistent if all versions can be accessed but only the newest version can be modified.

Similarly a data structure is Fully persistent if every instance can be both accessed and modified.

Persistence can be achieved by creating copy of the whole data everytime a change is made, but this is inefficient.

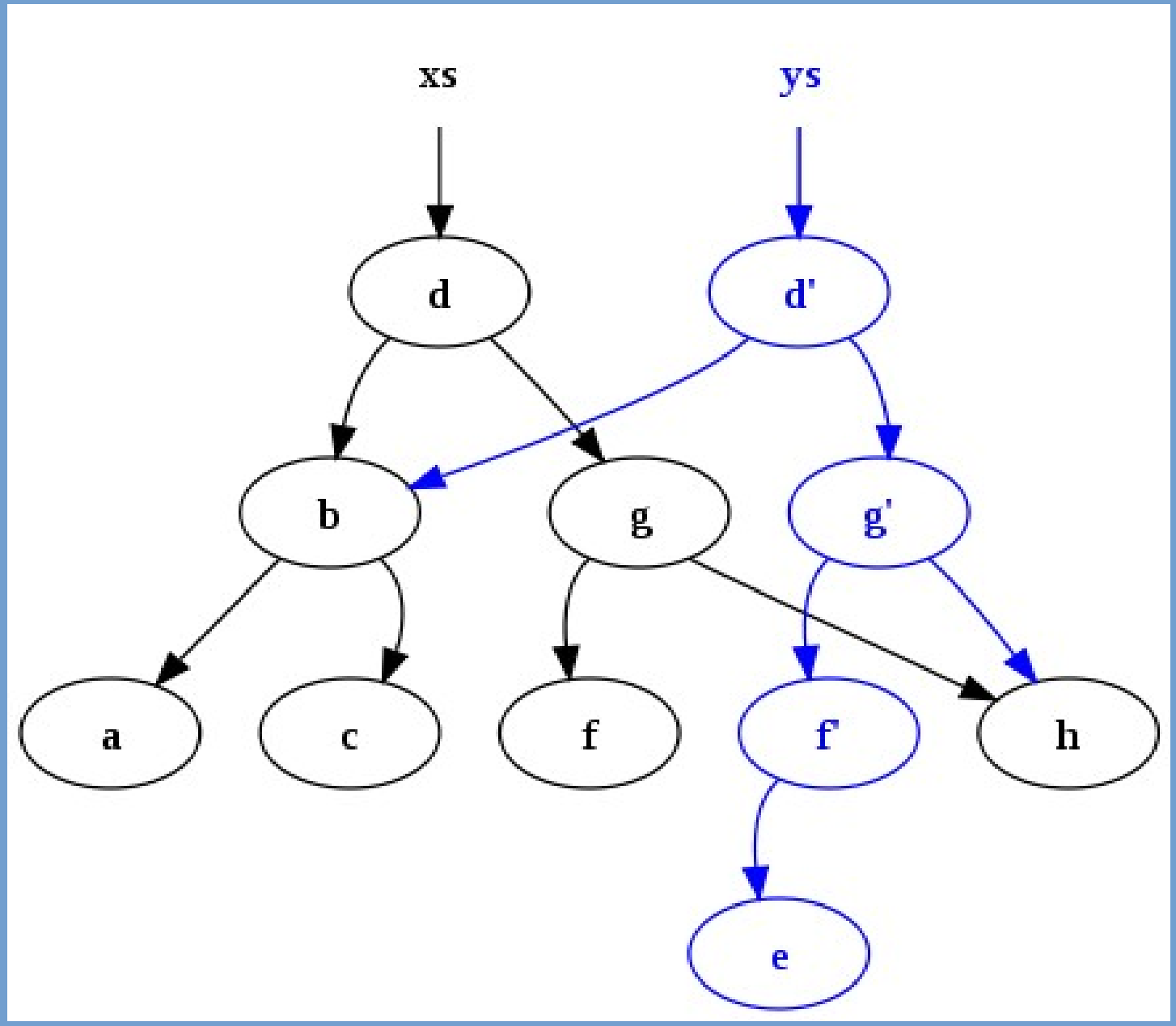
Our approach

Here we are implementing Fully persistent data structure on Binary Search Trees.

The nodes of the tree are files.

Any operation performed takes time in the order of the height of the tree{O(h)}.

We can access and make changes in the previous versions of the file in different branches.


xs
ys
d
d'
b
g
g'
a
c
f
f'
h
e

Insert and Delete

While inserting, we make a copy of all the nodes which are affected by the insertion in the current branch. This is done to ensure that the changes are only made to the current branch and it does not affect the other branches.

All the other nodes are reused in this way and memory is saved.

Finally when we reach the leaf, we insert the node there according to the BST property.

Similarly while deleting, a copy of all the affected nodes is made.

Project Structure

There is a “PBST” class object which represents the main tree.

It has an attribute “branches” where pointer to the root of every branch is stored.

The file display.py is used to run the GUI to run the various operations.

It is a menu driven program where we work on one branch at a time and can add, view, delete, search and edit files in that branch.

We can branch out from this branch or switch between the branches whenever needed.


PROJECT
VERSION CONTROL
MENU
List Files
View File
New File
Delete File
Edit File
List Branches
New Branch
Switch Branch
Exit
MADE BY
Yogesh Choubey : 17IT252
Dhruvik : 17IT225
Rakesh Pavan : 17IT154
Neeraj : 17IT226

THANK YOU