Software Methods and Tools

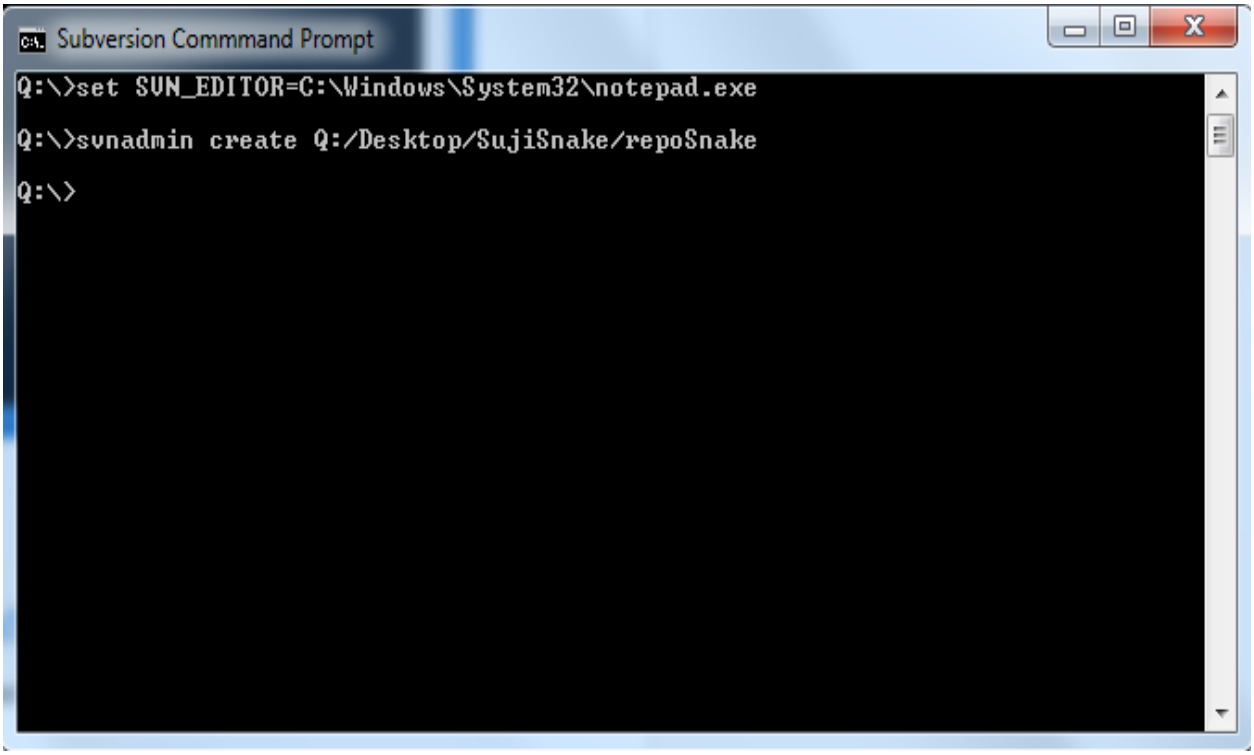
Assignment 7

1. Create a new repository, and create a project named “Snake” at the root of your repository. Then create the following folders under “Snake”: “trunk”, “branches”, and “tags”. Submit a screenshot that shows the content the “Snake” folder.

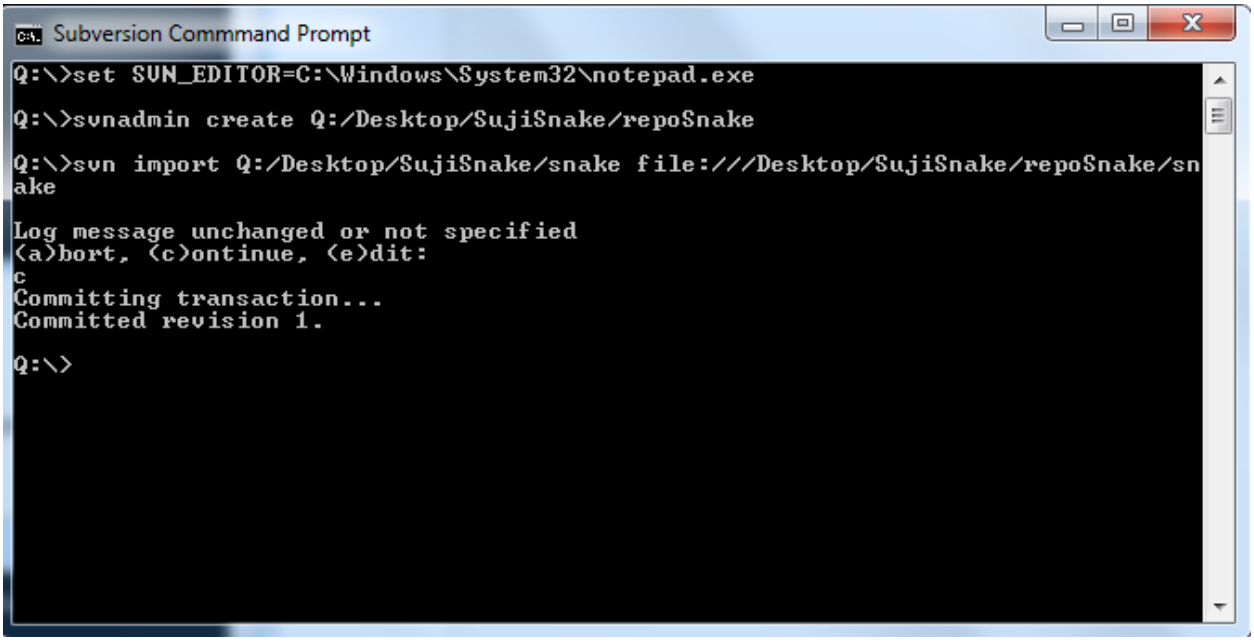
Command to make Windows’ NotePad application to edit the Subversion messages

“set SVN_EDITOR=C:\Windows\System32\notepad.exe”

Creating a Repository with name repoSnake in folder \\kc.umkc.edu\kcusers\home\s\spb4b\Desktop\SujiSanke.

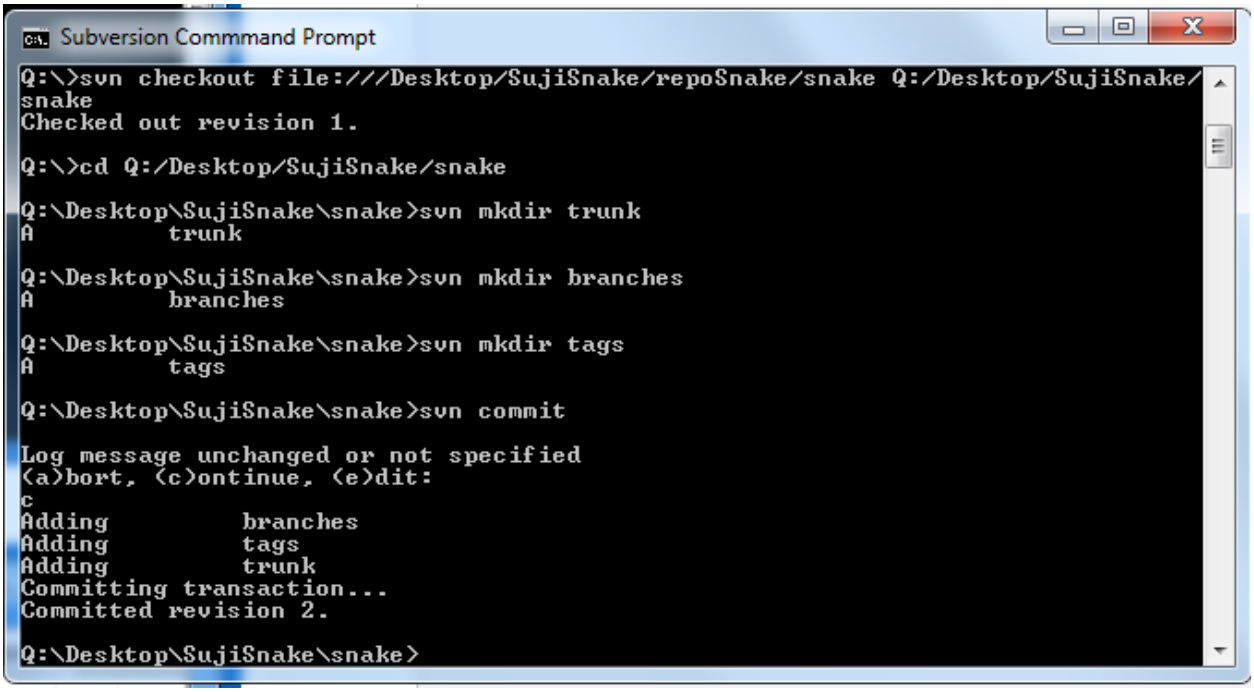


Creating new folder Snake at the root of Repository repoSnake and creating a project into repository,

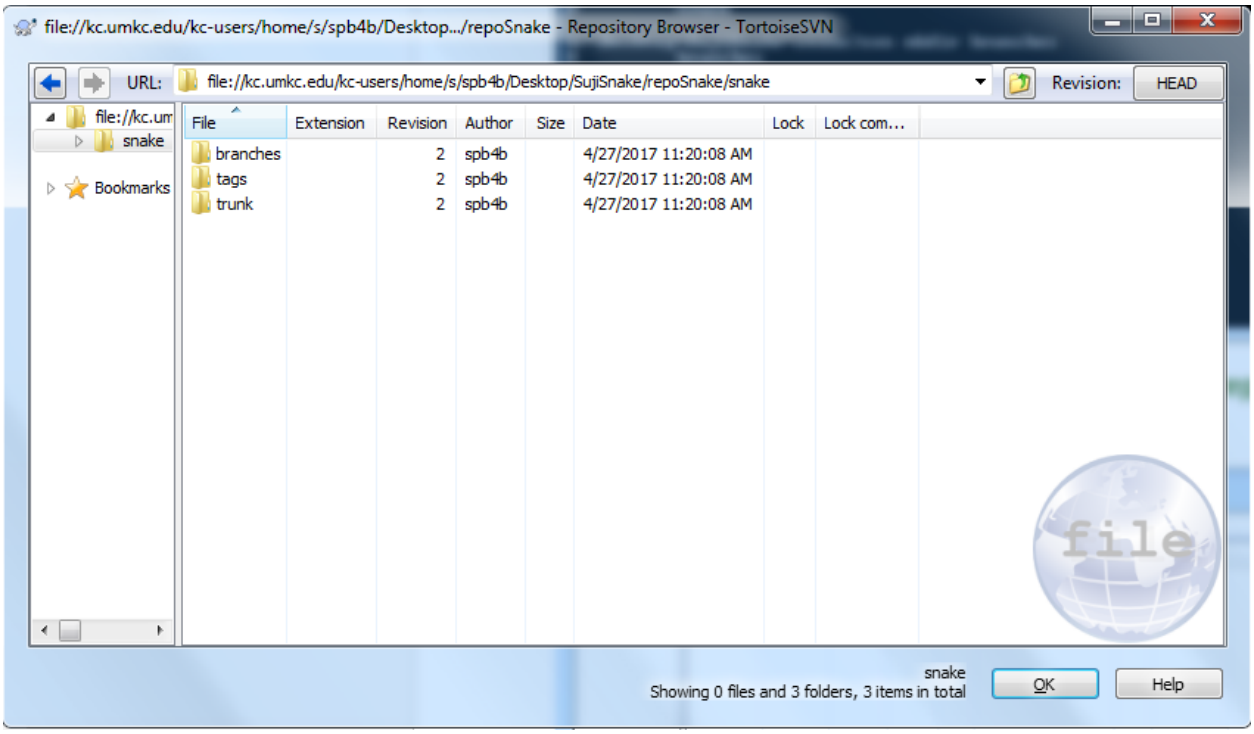


THE REVISION NUMBER “1” is created in the repository.

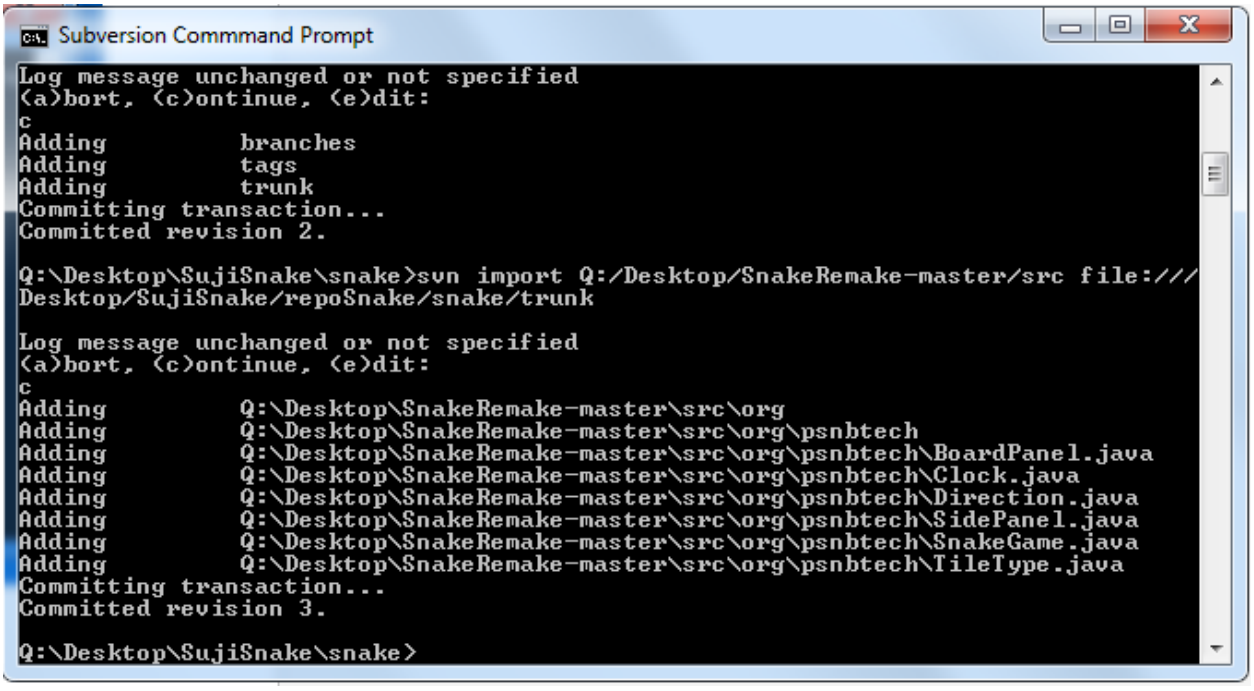
Creating the folders under “Snake” : “trunk”, “branches”, “tags” and verifying through SVN if these folders exist in repository.



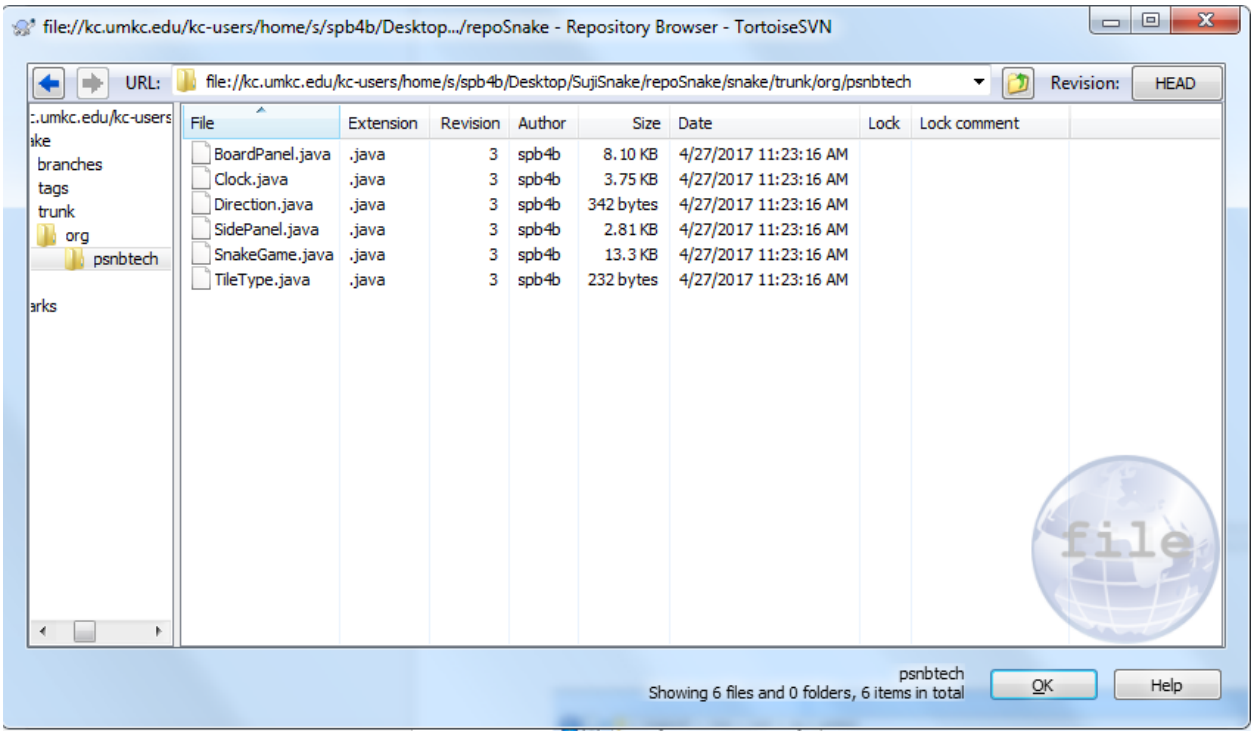
THE REVISION NUMBER “2” is created in the repository.



2. Download the source code from https://github.com/PSNB92/SnakeRemake. Put the unzipped source code (i.e. *.java) of the Snake application into the "trunk" folder of the repository. Keep the original file structure (i.e. org.psnbtech.*). Submit another screenshot that shows the content of the source code folder in the repository.



THE REVISION NUMBER “3” is created in the repository.

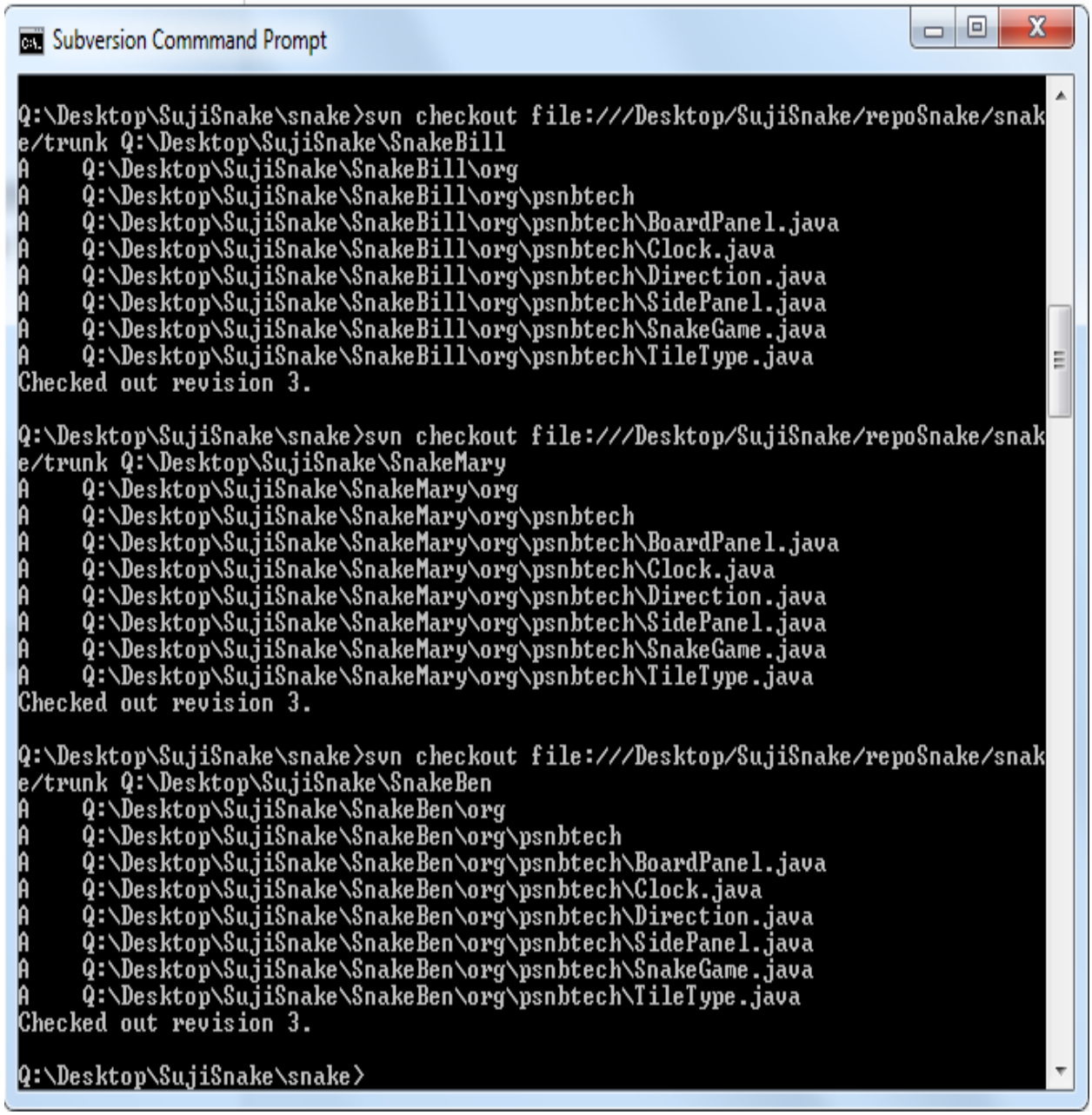


3. Create three working copies of the Snake application. You can do this by checking out the project “Snake/trunk” into three different folders, one for each developer (e.g. Bill, Mary, and Ben). The folders you created should be named as SnakeBill, SnakeMary, and SnakeBen.

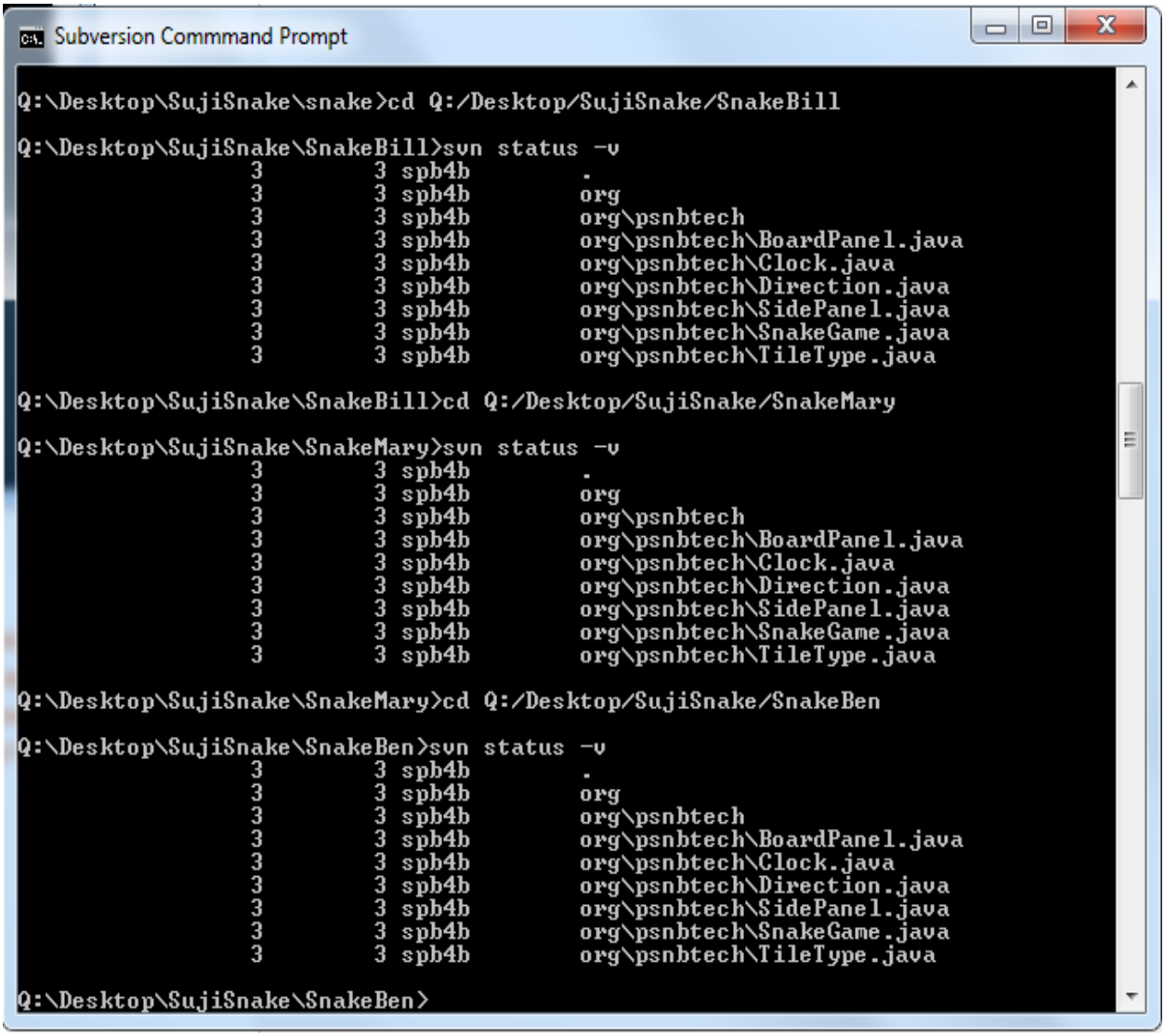
Creating the three-new working copy as

- SnakeBill - Q:/Desktop/SujiSnake/SnakeBill
- SnakeMary - Q:/Desktop/SujiSnake/SnakeMary
- SnakeBen - Q:/Desktop/SujiSnake/SnakeBen

Checkout the repository into theses 3 working copy.



4. Simulate the following events by performing the corresponding operations. After each event: (1) check the revision number of the SnakeGame.java file in the repository and in each working copy, and fill in the table below; (2) describe what problems (i.e. conflicts) you had in each event, and how you resolved them. (3) submit a screenshot showing the revision numbers of the Java files in Mary's working copy.

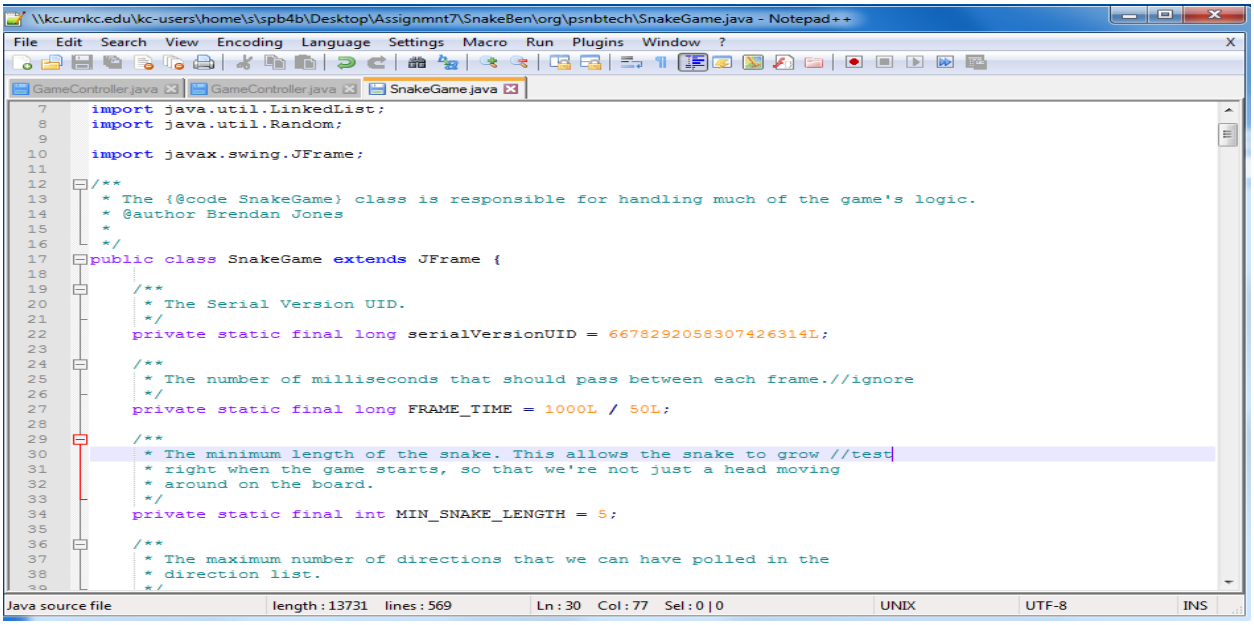


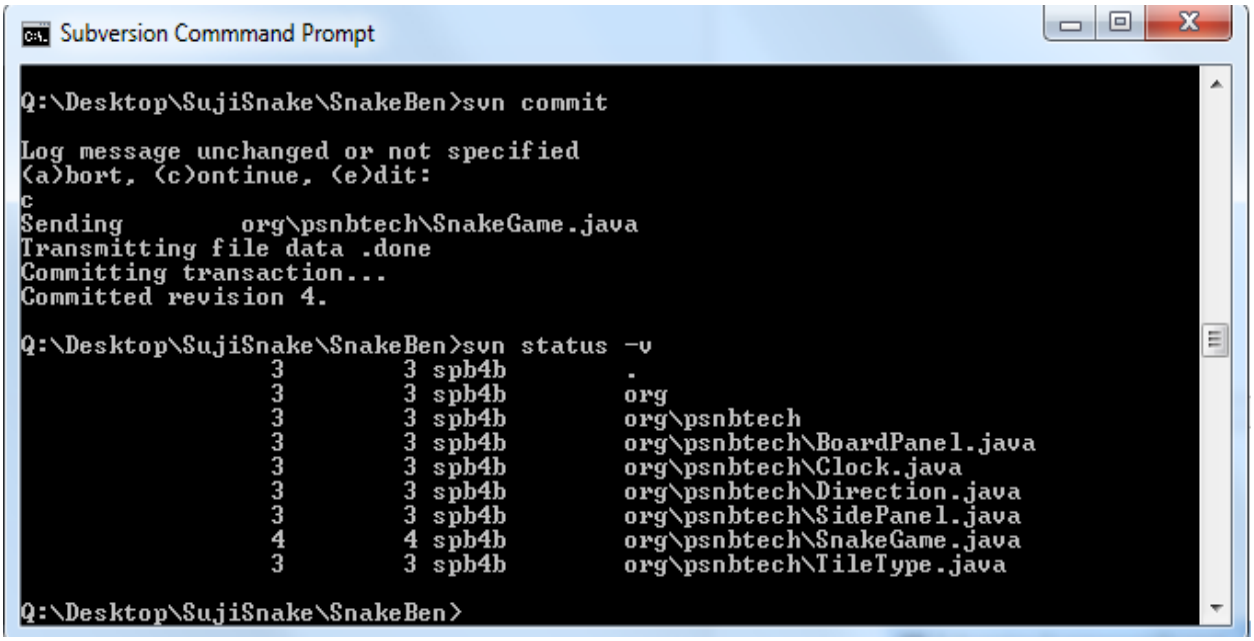
THE REVISION NUMBER IN ALL THE WORKING COPY IS “3”.

EVENT #1:

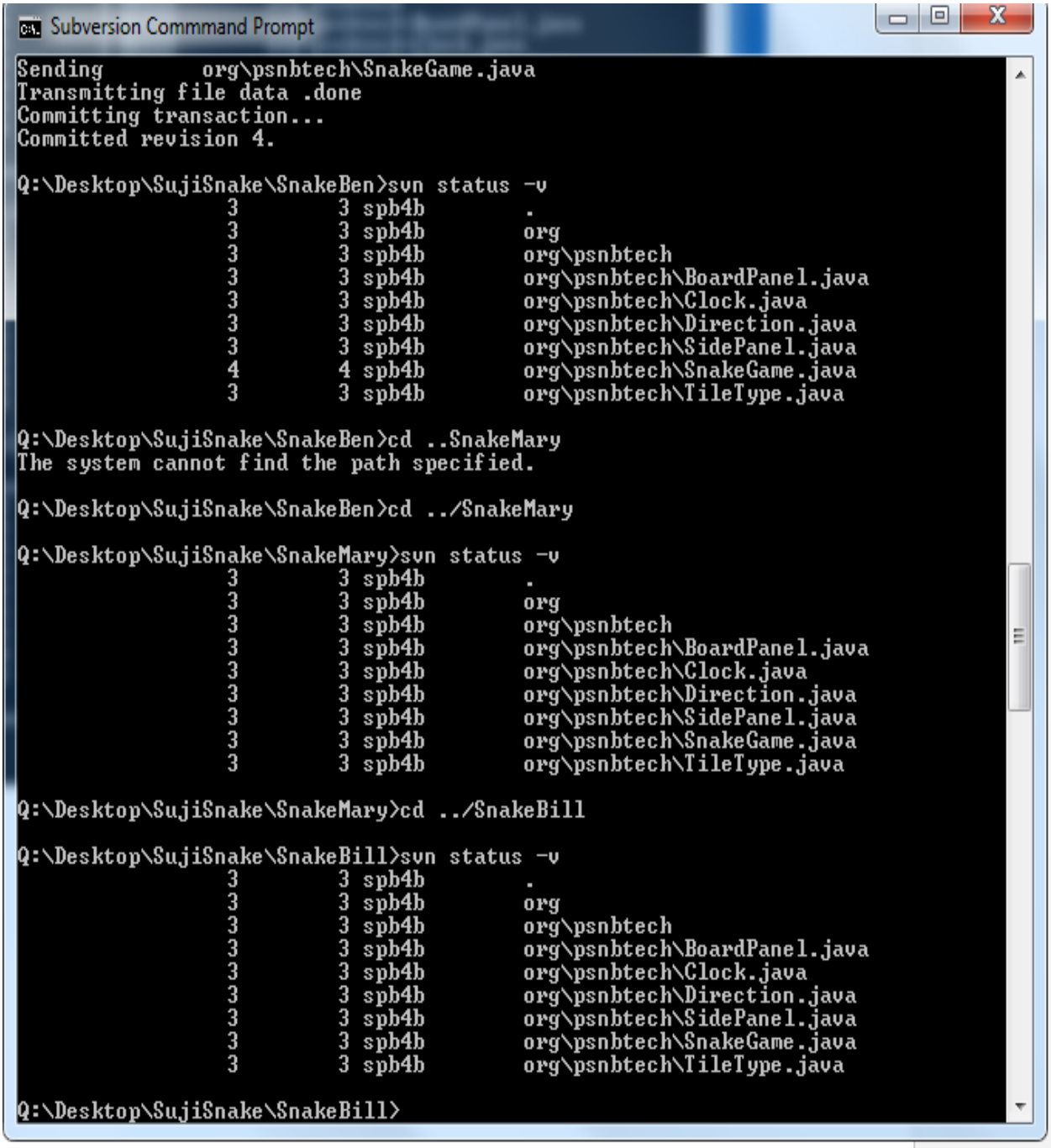
Ben made some changes to SnakeGame.java: (1) added a comment “//igore” at the end of Line 25, and a comment “//test” at the end of Line 30. Then Ben committed his changes to the repository.

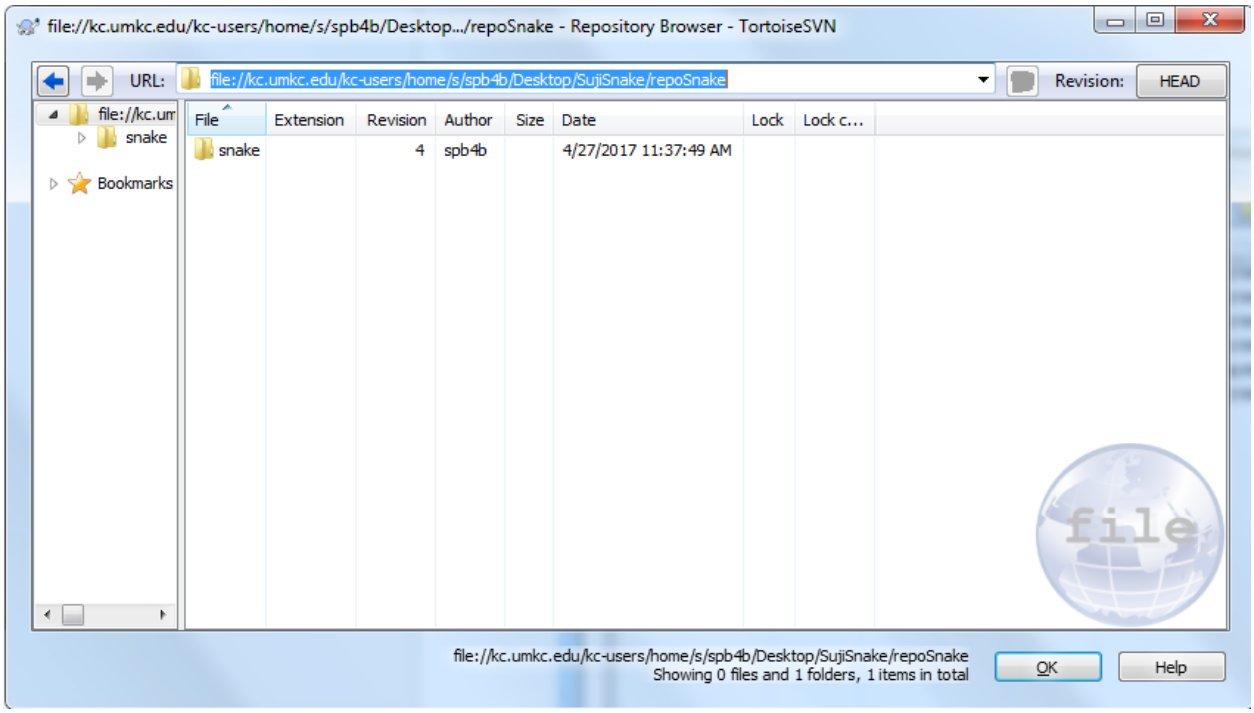
Modifying Code:





THE REVISION NUMBER OF REPOSITORY IS “4” AFTER COMMIT.





Problems Faced: None

Revision Numbers of SnakeGame.java:

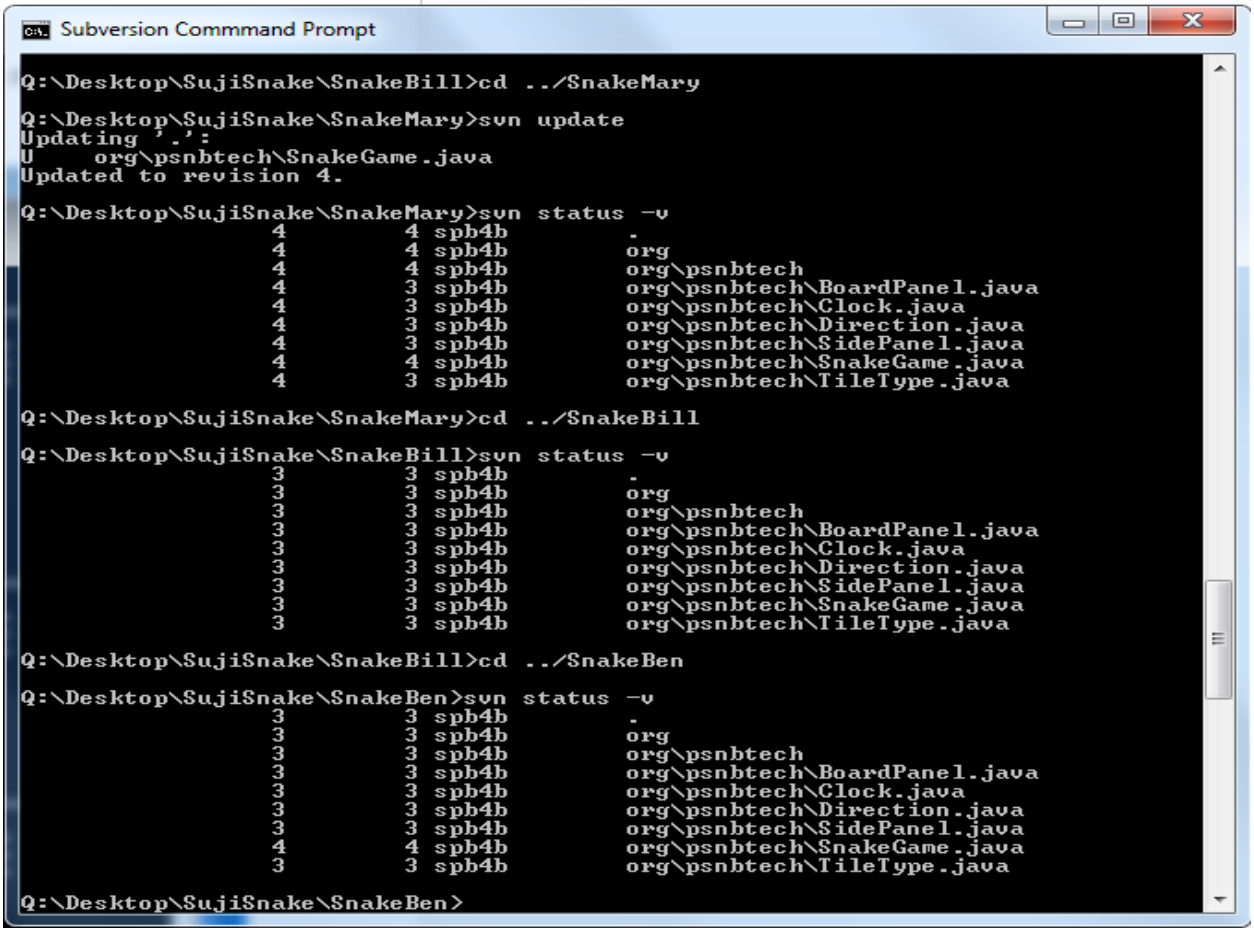
Event	Repository	Bill	Mary	Ben
1	4	3	3	4

Description: As only Ben has committed his modified code, so the repository and ben revision number became 4. Still Bill and Mary did not take their update of the repository their revision number would be still 3.

EVENT #2:

Mary synchronized her working copy with the repository.

Mary SVN Update:



Problems Faced: None

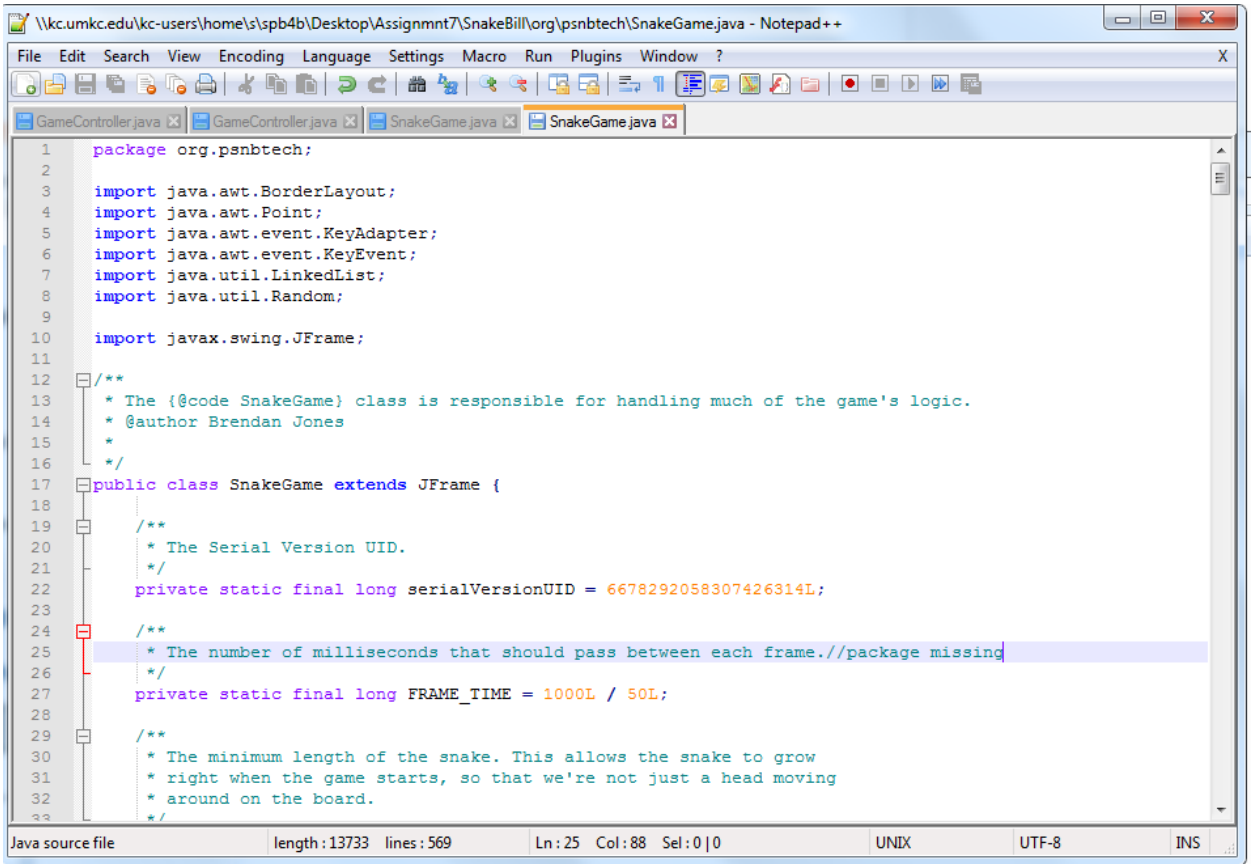
Revision Numbers of SnakeGame.java:

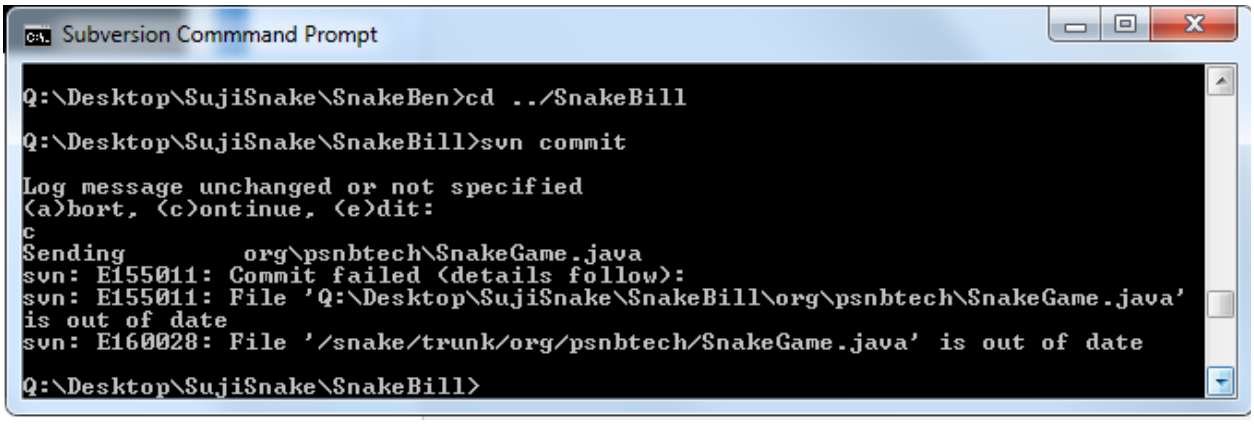
Event	Repository	Bill	Mary	Ben
2	4	3	4	4

Description: As Mary took the update her revision number changes to 4 after synchronization with the parent repository.

EVENT #3:

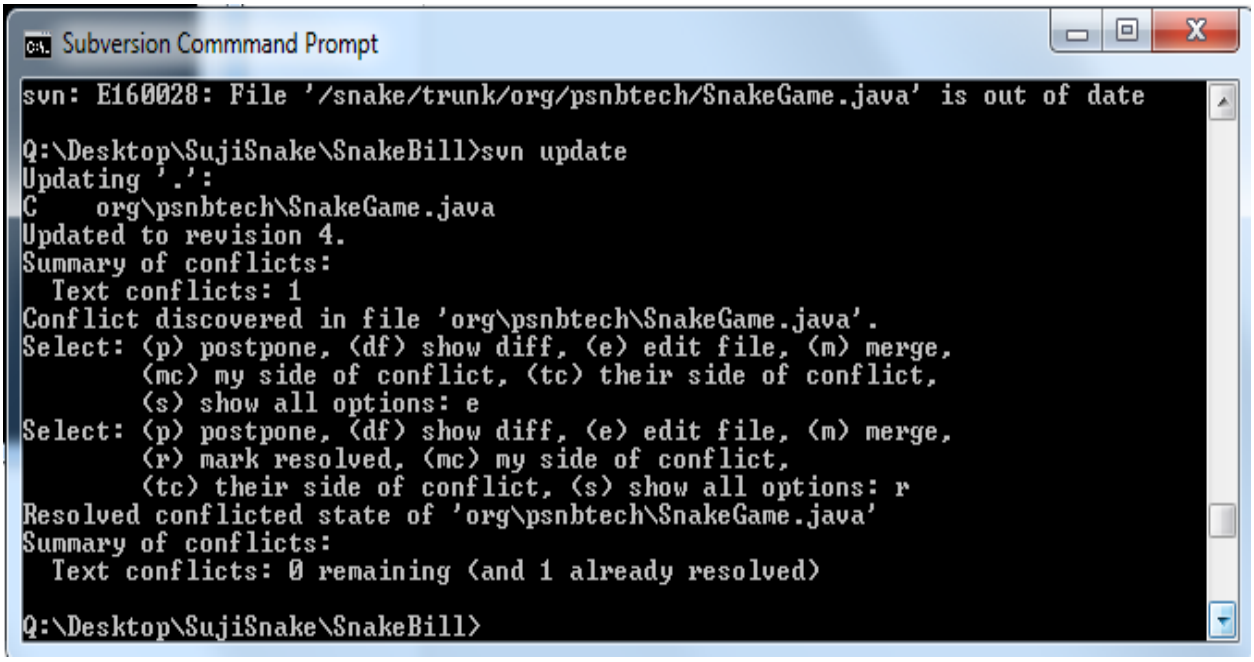
Bill made a change to SnakeGame.java: added a comment "//package missing." at the end of line 25. Bill resolved the conflicts, and committed his change to the repository.

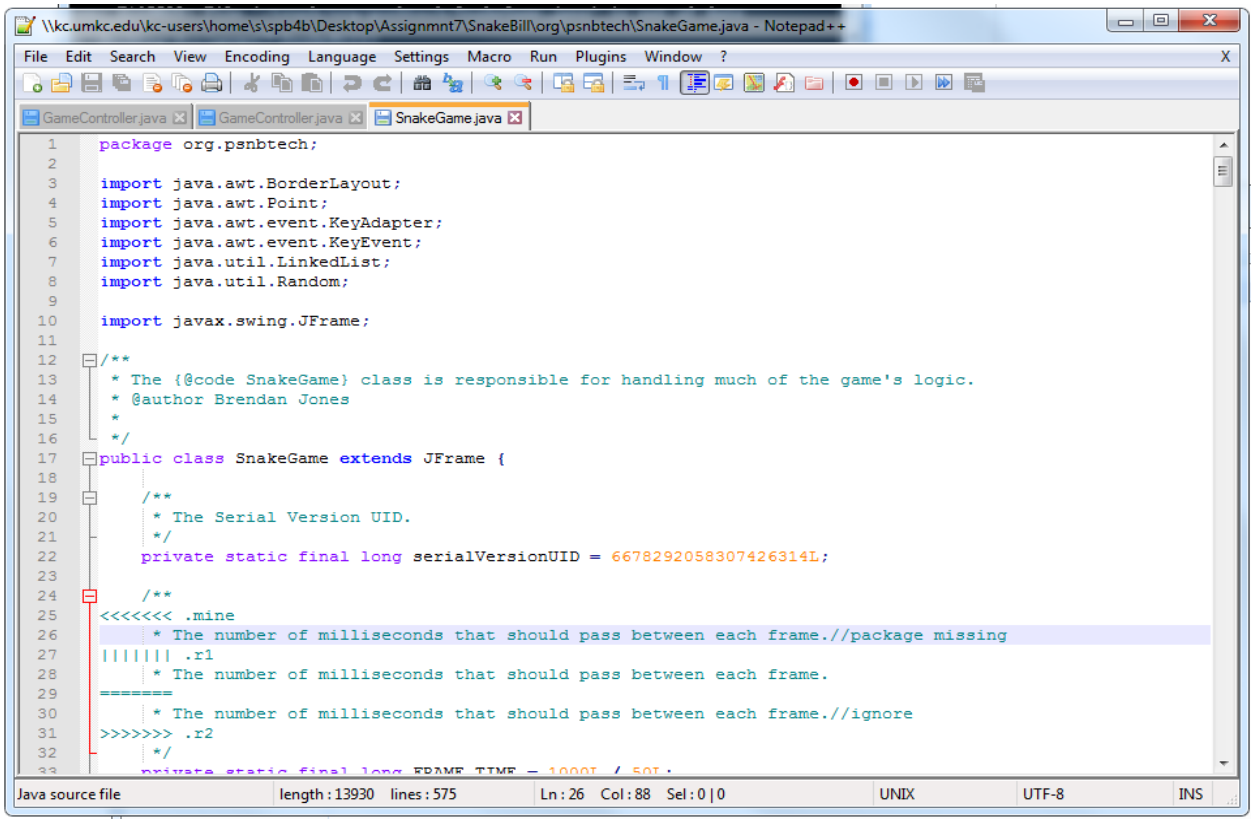




Problems Faced: Conflict error due to out of date.

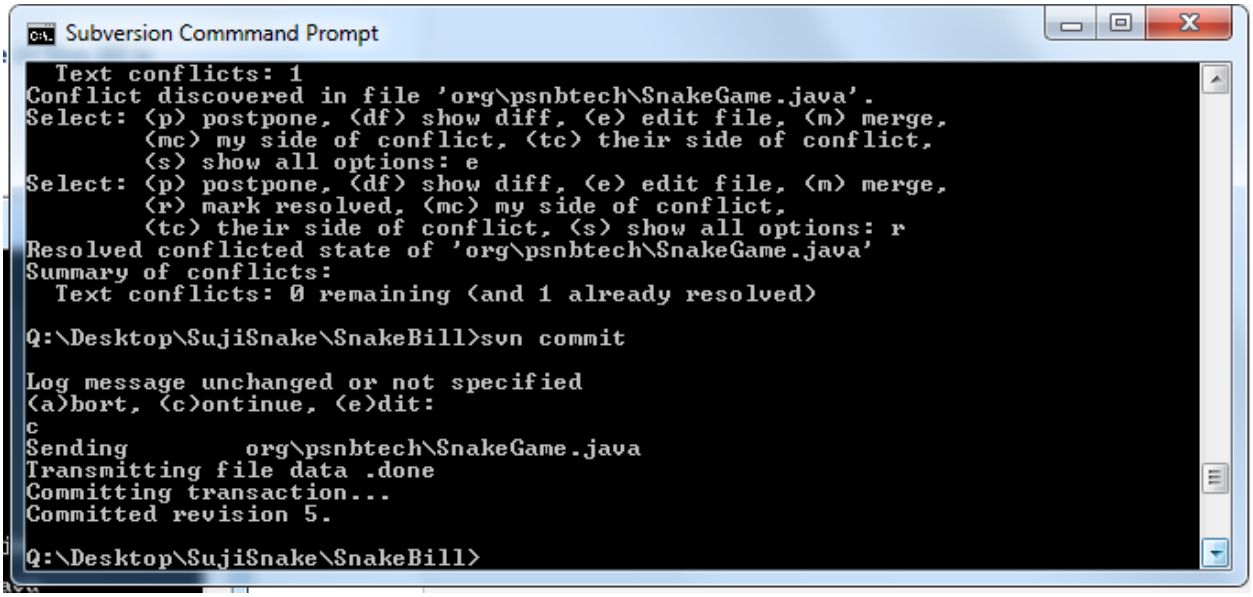
Solution: Update the working copy and edit and mark as resolved.



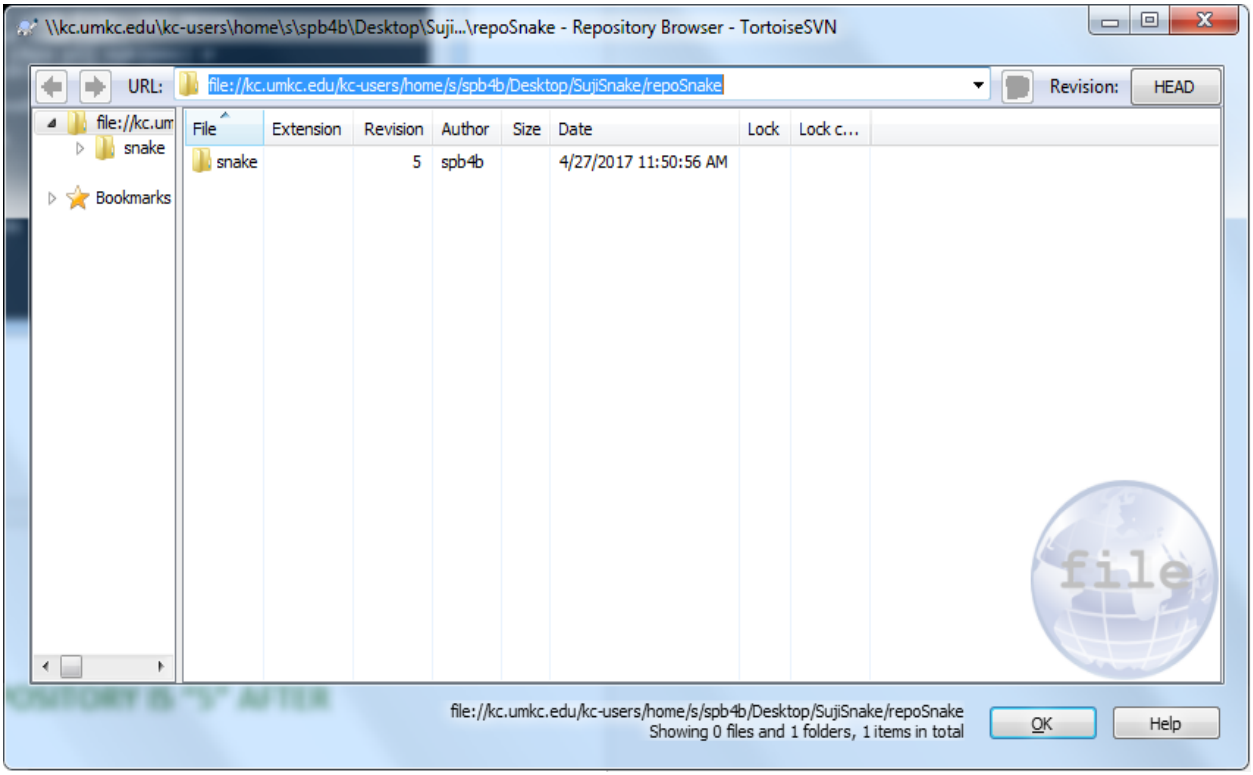


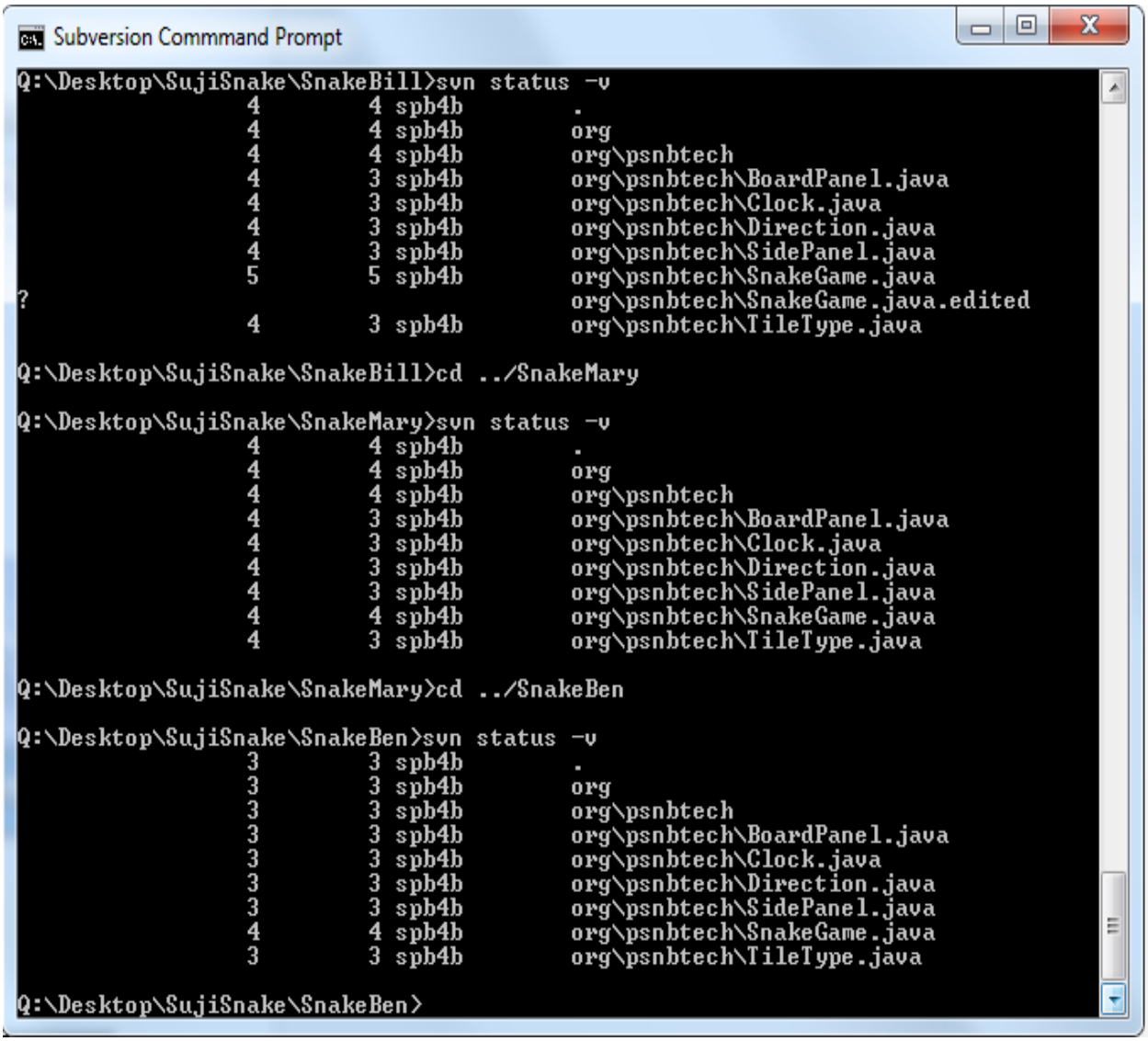
The code in the line 25 is updated with latest Bill code and the Ben update in line 30 is synchronized.

Commit Code and check for version number:



THE REVISION NUMBER OF REPOSITORY IS “5” AFTER COMMIT.





Revision Numbers of SnakeGame.java:

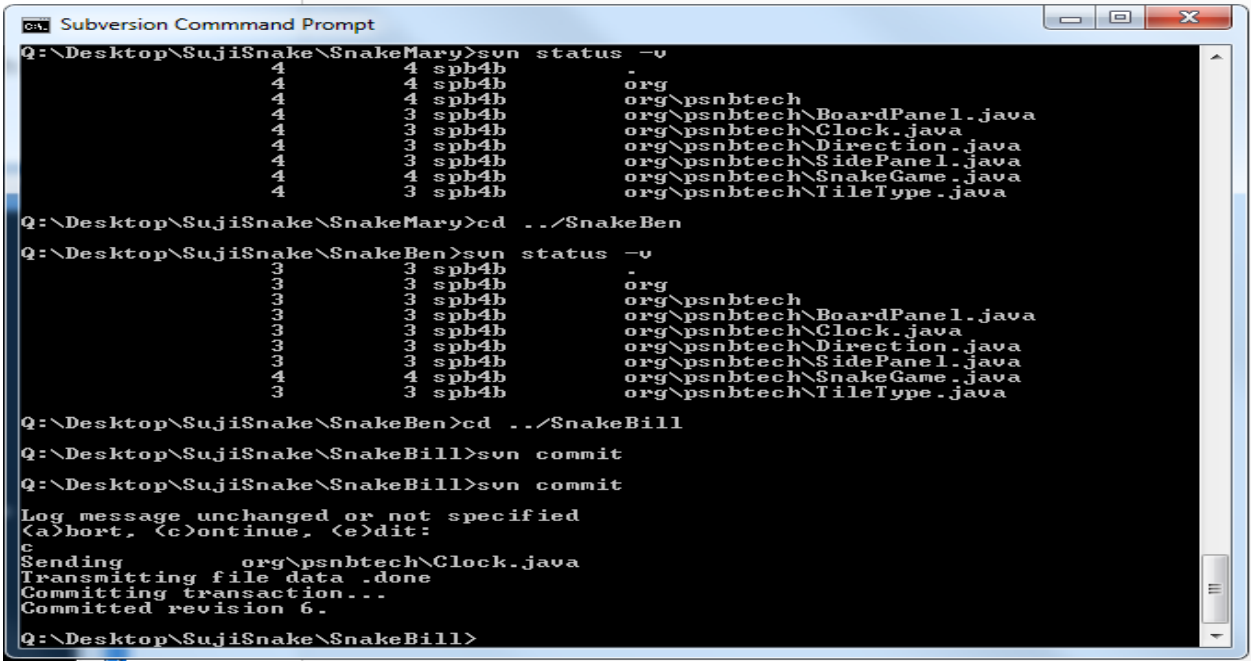
Event	Repository	Bill	Mary	Ben
3	5	5	4	4

Description: Ben tried to commit his code without taking an update then error has occurred, so he should take an update and edit the code and mark as resolved before committing his latest code. The error is occurred so the previous modification by others is not deleted.

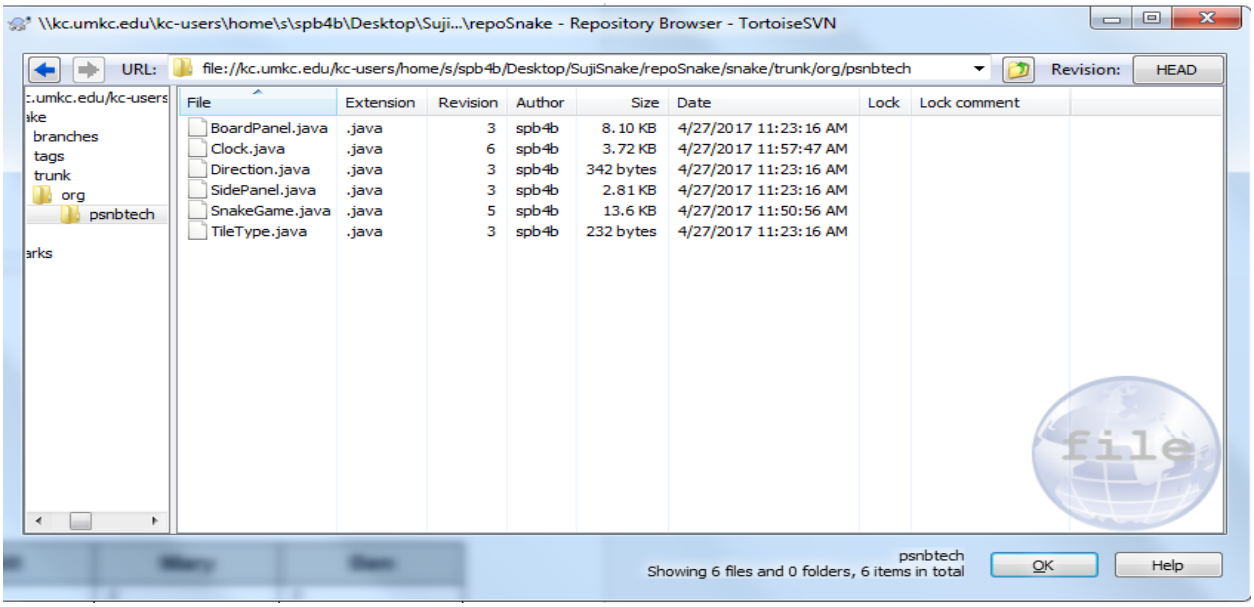
EVENT #4:

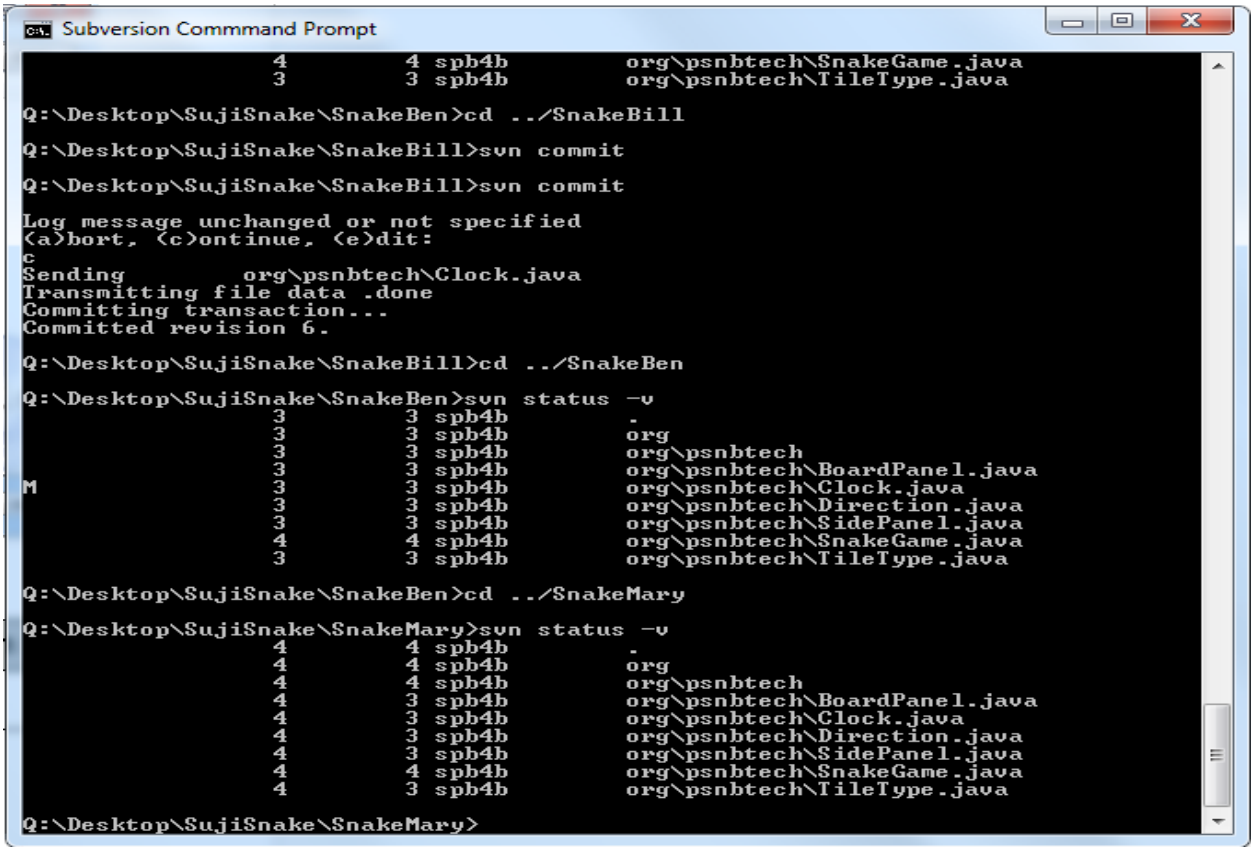
Bill changed the file, Clock.java (e.g. removed the fifth line of code). He then committed his change to the repository.

Bill removed line 5 of Clock.java and commit.



THE REVISION NUMBER OF REPOSITORY IS “6’





Problems Faced: None

Revision Numbers of SnakeGame.java:

Event	Repository	Bill	Mary	Ben
4	5	5	4	4

Revision Numbers of Clock.java:

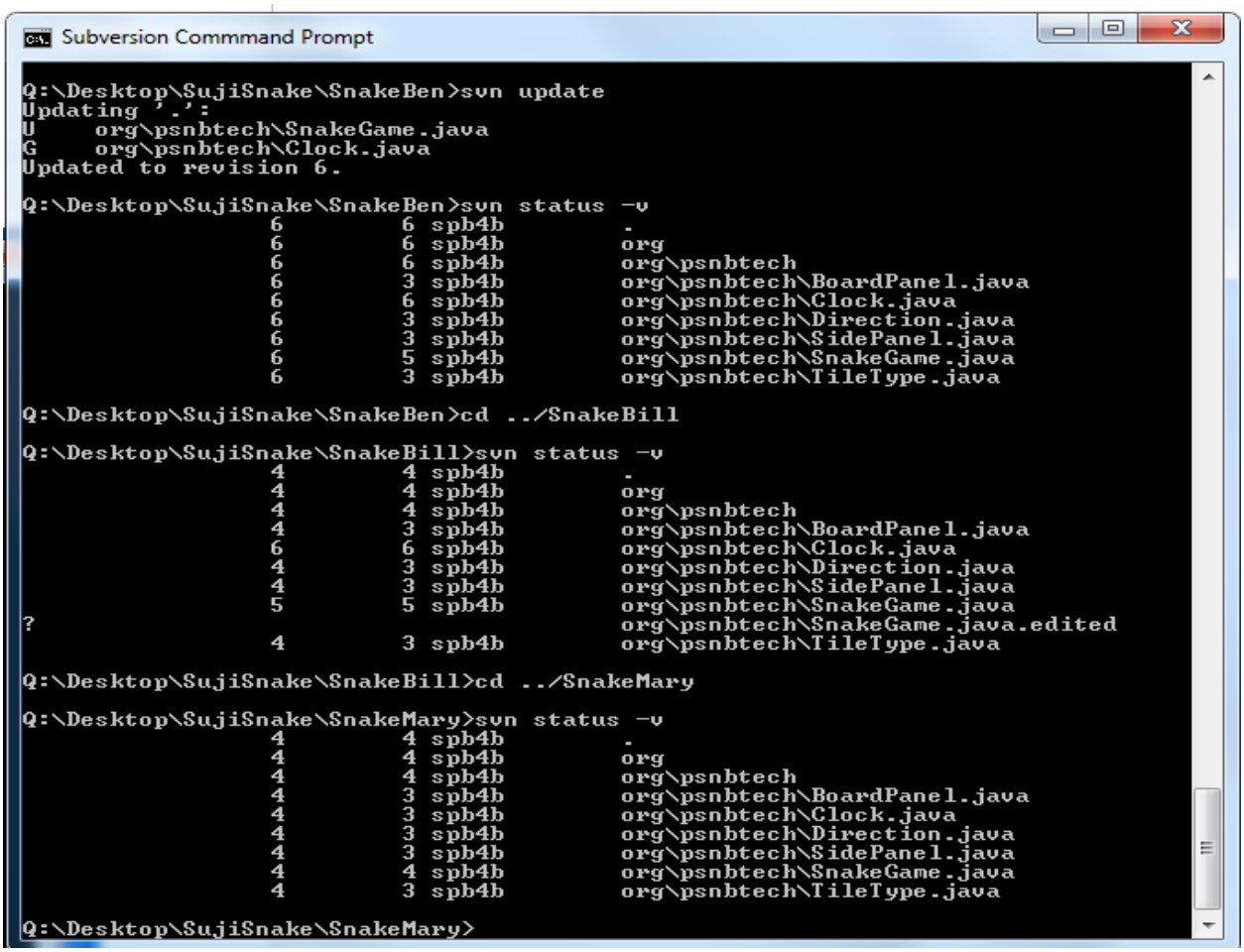
Event	Repository	Bill	Mary	Ben
4	6	6	3	3

Description: Bill modified the Clock.java code so the revision number is incremented for repository and Bill working copy. But the SnakeGame.java revision number is not incremented.

EVENT #5:

Ben synchronized his working copy with the repository.

Ben Update his Copy:



Problems Faced: None

Revision Numbers of SnakeGame.java:

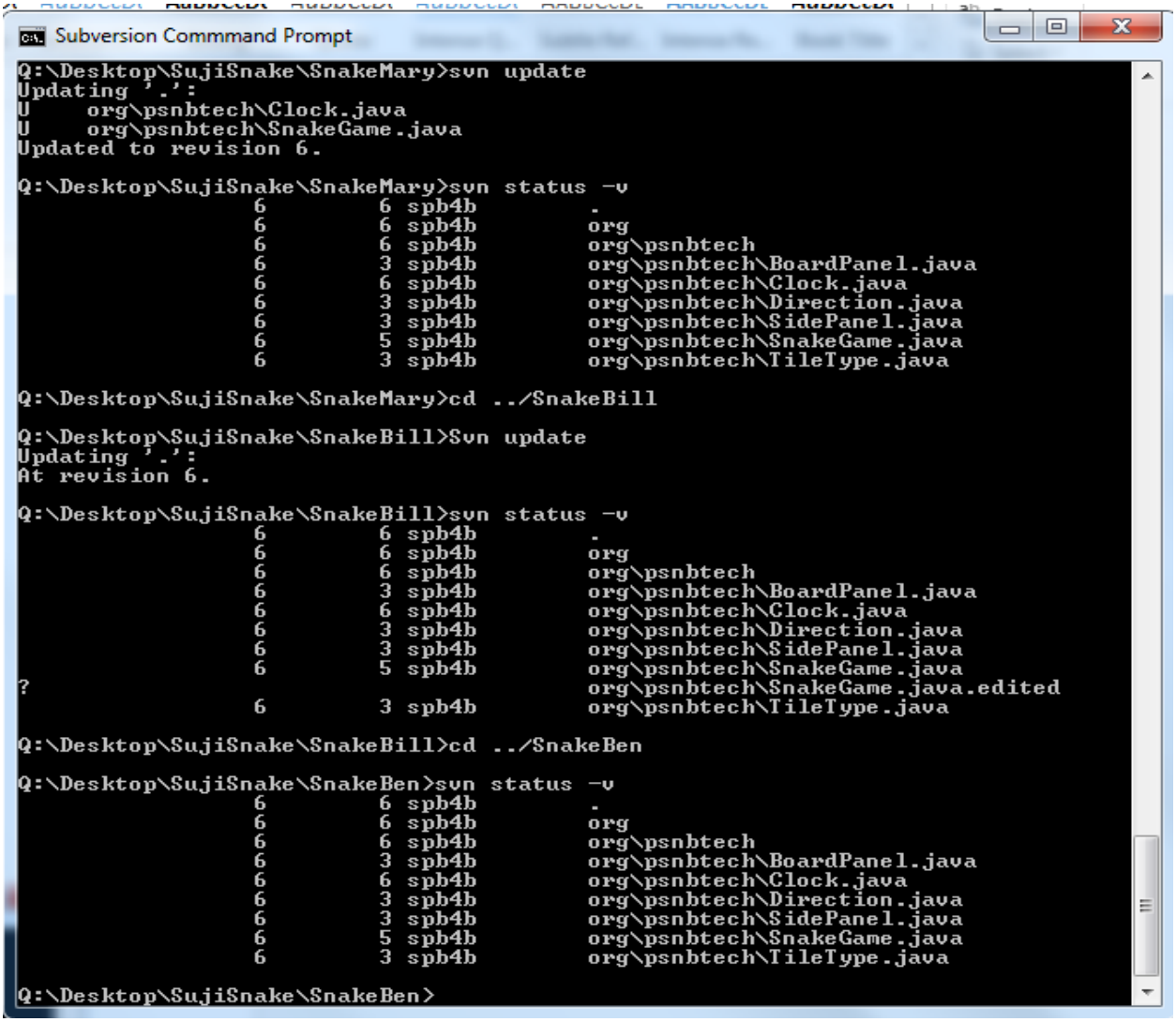
Event	Repository	Bill	Mary	Ben
5	5	5	4	5

Description: Ben working copy is synchronized with revision number of repository.

EVENT #6:

Mary and Bill both synchronized their working copies with the repository.

Mary and Bill Update their Copy:



Problems Faced: None

Revision Numbers of SnakeGame.java:

Event	Repository	Bill	Mary	Ben
6	5	5	5	5

Description: Bill and Mary have the latest copy of project after the update.

COMANDS USED:

Command	Use
set SVN_EDITOR=C:\Windows\System32\notepad.exe	To use the notepad as editing source
svn admin create Q:/Desktop/repo	To create the central repository
svn import Q:/Desktop/Snake file:///Desktop/repo/Snake	To import project into central repository
svn mkdir directoryname	To create a directory (in the specified folder use cd… /Snake)
svn checkout file:///Desktop/repos/snake/trunk Q:/Desktop/SnakeBill	To checkout
svn status -v	To check for status
svn commit	To commit code to repos
svn update	To update the code to the revision number of repos.

Observation Notes:

1. **Its good practice to take an update before modifying the code.**
2. **When an old version working copy is modified and committed then it throws an error.**
3. **A new edited copy is created when merging with an new version copy.**
4. **Only the version number of the updated file is incremented and for other code it's not incremented.**