Linking files lab answer

Answers for Lab 4

On the top of all your documents, always include these items:

your name: xxxxxx
date: xx/xx/xx
Lab number 4

This Week lab with extra points!!!

You will need to post a message in the "Discussion board" and request for a partner. You will plan with your partner when to complete this activity but remember that it has to be done during or before the official lab hours. You can catch 20 pts if you finish it correctly.

If answer is complete and accurate, each member of the two person team will get 20 bonus points.

You are to create a hard link to one of your existing files on someone else's directory (or vice versa). In other words, you know that you can link a file within your own directories, but you can also have a link to one of your files on other areas of the unix system as long as you have permissions to write to that directory (in this case, your partner).

Create a subdirectory called temp where you can place this temporary link.

Remember that you do not link a file to another file. You create a hard link to an existing file.

So, *user A* has **file1** that he/she wants to give access to *user B*. *User B* has to open certain directory permissions, for this to happen, then *User A* can create the link on user B directory (user A NEVER goes to user B directories typing cd), but if permissions are opened correctly all the commands will be done from user A directory without shell error messages such as “can not access..”.

The question? what did you have to do for this work? Describe ALL the steps labeling. Be very precise on these steps. A person should be able to take step by step described and accomplish the task of creating a hard link. Make sure to be precise. I should be able to follow these steps and create the link.
step 1. ...
step 2.
step 3..

Upload your answer to the Dropbox under “Lab 4 submission” with your name and partner (if you have one) before the end of the official lab time.

In Unix, what is a hard link?

A hard link is essentially a label or name assigned to a file. Conventionally, we think of a file as consisting of a set of information that has a single name. However, it is possible to create a number of different names that all refer to the same contents. Commands executed upon any of these different names will then operate upon the same file contents.

To make a hard link to an existing file, enter:

```
ln oldfile newlink
```

Replace oldfile with the original filename, and newlink with the additional name you'd like to use to refer to the original file.

This will create a new item in your working directory,newlink, which is linked to the contents ofoldfile. The new link will show up along with the rest of your filenames when you list them using the ls command. This new link is not a separate copy of the old file, but rather a different name for exactly the same file contents as the old file. Consequently, any changes you make to oldfile will be visible in newlink.

You can use the standard Unix rm command to delete a link. After a link has been removed, the file contents will still exist as long as there is one name referencing the file. Thus, if you use the rm command on a filename, and a separate link exists to the same file contents, you have not really deleted the file; you can still access it through the other link. Consequently, hard links can make it difficult to keep track of files. Furthermore, hard links cannot refer to files located on different computers linked by NFS, nor can they refer to directories. For all of these reasons, you should consider using a symbolic link, also known as a soft link, instead of a hard link.

If answer is complete and accurate, each member of the two person team will get 20 bonus points.

1. *create another dir in your home directory like link_dir*
2. *the user receiving the link(USERB) needs to open rwx permission for link_dir*
3. *the user receiving the link needs also to open X in the home directory for group so USERB home directory will show drwx—x--- (just X. You can open the others, but opens vulnerability to other files in your home directory)*
4. *userA needs to change permission of the file to be linked to rw-rw----(this way USERB can read/write to the linked file)*
5. *USERA then will create the link by doing*
   do ln ~usera/link_dir/my_originalfile ~userb/link_dir/linktouserafile (linktouserafile file will not exist, USERA is creating this link)The inode for the file and the link will be the same, you can check by doing ls -i

Notes

When the owner of the original file "my_originalfile" changes the permission, linktouserafile will also have those permissions changed.

You DO NOT link a file to another existing file

Why would you would link files?

for instance, you can have a homepage file .htm where multiple users need to update it to be published, like different departments that have a centralized calendar.

Programmers that have programs to write and work on it at different times or locations.

You can have for instance a file called calendar and subdirectories where you have different projects like /intel /microsoft and just links in them, so when you make a change to the link of the calendar file the changes will be in all dirs.

or a shared calendar where everyone make entries.