10. Tenth Assignment – Multi-level Undo

Sometimes, on rare occasions, people do things they regret on Facebook. In this assignment you will implement an “undo” option do handle such circumstances. You will need to keep track of all of the “undo-able” actions that occur within your application. An action is undo-able if it changes the data. For instance, “list users alphabetically” is not undo-able, while “add friend” should be. You will then need an “undo” menu item. When the user chooses this option, the program will perform the reverse of whatever the last action was. For instance, if the last action was Alice adding Bob as her friend, undoing this would cause the program to prompt for Alice’s password and then remove Bob from her friends list. Conversely, if the most recent action was for Charlie to de-friend Alice, undoing this would prompt for Charlie’s password and then add Alice back as one of his friends. If there are no actions left to undo, display an error message to the user. The options that could be considered undo-able are: add a user, delete a user, add friend, remove friend, and liking something. At a minimum, you are required to implement undo functionality for the add a friend and remove a friend option. *You will receive one extra credit point for any of the other three options that you make undo-able.* Similar to previous assignments, you need to become comfortable with choosing the appropriate data structure for a particular task. With this in mind, **you are required to have a comment somewhere in your Facebook class that indicates which data structure you have used to store the undo-able actions and explains why you have chosen it.** Begin this comment with the word EXPLANATION (in all caps) so that I can search for it when grading your work.

You will be graded according to the following rubric (each item is worth one point):

- The driver menu contains the new undo option
- The program prompts for the password of the relevant user (the one who performed the most recent undo-able action)
- The action is undone only if the password is correct for the relevant user
- The add friend action is undo-able in your program
- The remove friend action is undo-able in your program
- If there are no undo-able actions left, an error message is displayed
- The choice of data structure to store the undo-able actions is logical and well-defended in a comment beginning with EXPLANATION
- The program compiles
- The program runs
- The program is clearly written and uses standard coding conventions
- ***Note:*** *If your program does not compile, you will receive a score of 0 on the entire assignment*
- ***Note:*** *If you program compiles but does not run, you will receive a score of 0 on the entire assignment*
- ***Note:*** *If your Eclipse project is not exported and uploaded to the eLearn drop box correctly, you will receive a score of 0 on the entire assignment*

- ***Note:*** *If you do not submit code that solves the problem for this particular assignment, you will not receive any points for the program’s compiling, the program’s running, or following standard coding conventions.*