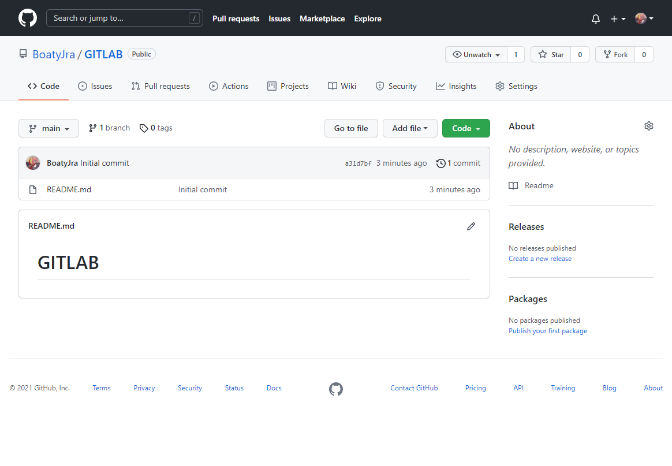
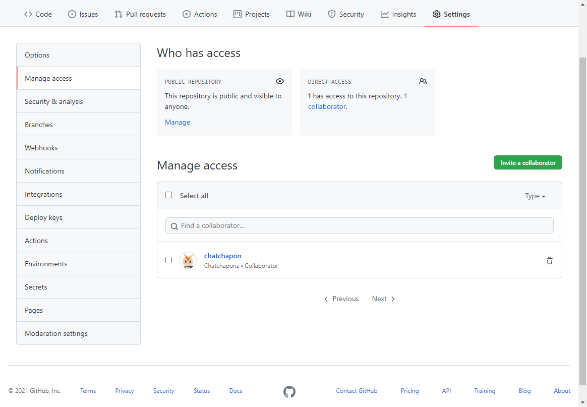
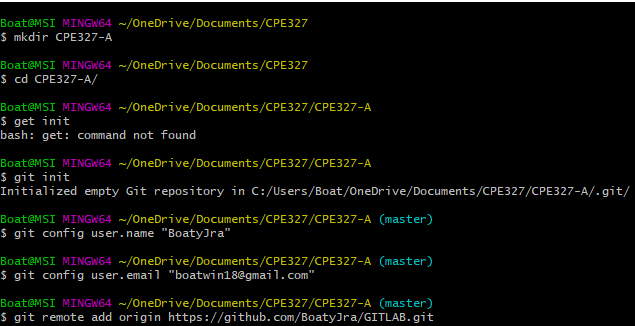
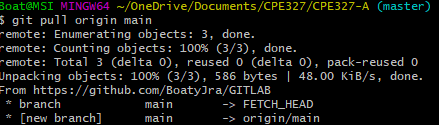
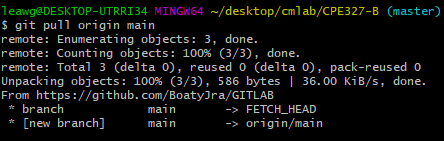
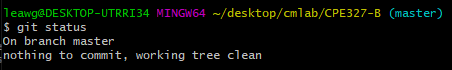
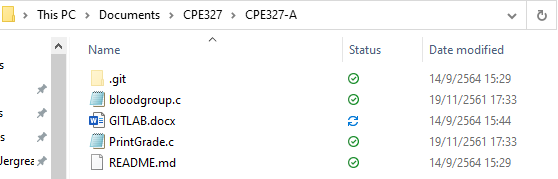
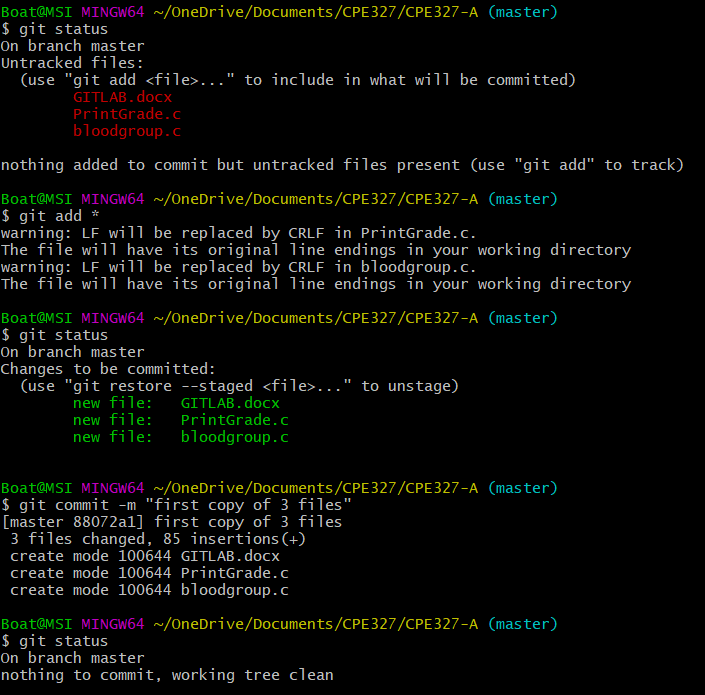
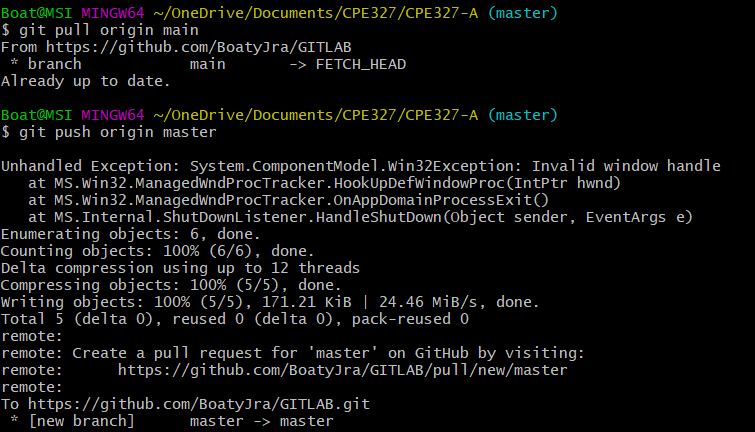
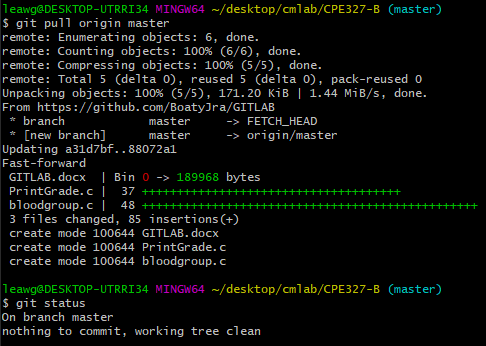
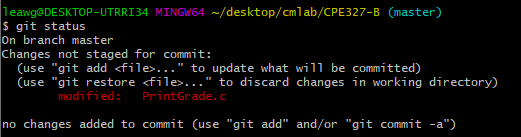
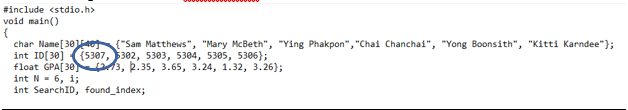
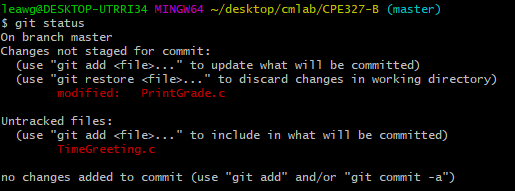
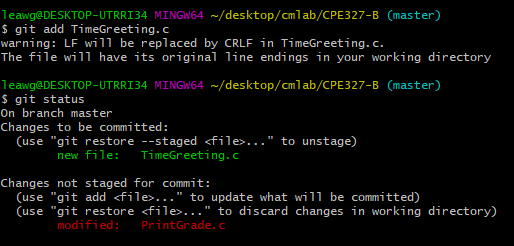
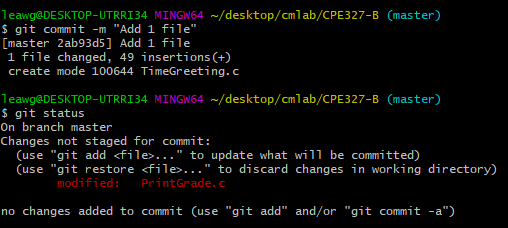
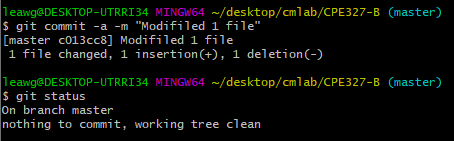
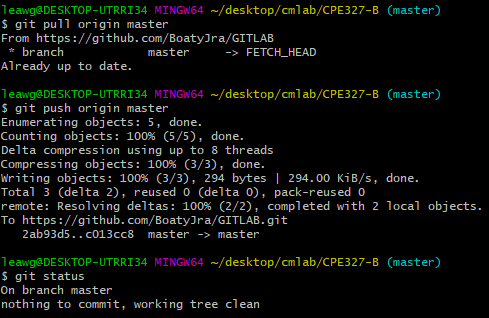
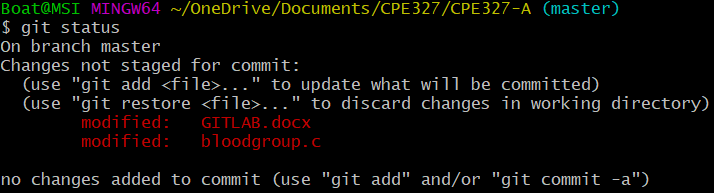
1. Unzip the lab zip file. The original source files can be found under src. The new file TimeGreeting.c is located in NEW.
2. Create a blank repository in GitHub under User A’s account called “GITLAB” with 1 Readme file. Print hub status. Both users A and B will use this repository. Add User B as collaborator to this repository
3. In some path create a folder called “CPE327-A” for User A. Make this a git repository and pull data from GitHub. Show status.  
     
   
4. In some path create a folder called “CPE327-B” for User B. Make this a git repository and pull data from GitHub. Show status.  
     
     
     
   
5. User A. Use file manager to put two C programs (bloodgroup.c and PrintGrade.c), and one MS Word document into CPE327-A  
   
6. User A. Add all files to git. Then commit. Show Status.  
   
7. User A. Push your local repository to the GitHub repository. (You should always pull before you push). Show screen  
   
8. User B pulls data from GitHub. Show status.  
   
9. User B change 1 function in PrintGrade.c. Print local status User B.  
     
   
10. User B create a new C program TimeGreeting.c. Do not put into Git yet. Print local status User B.  
    
11. User B adds TimeGreeting.c to Git control (git add). Print local status User B.  
    
12. User B commits changes. Print local status User B.  
      
    
13. User B pushes changes to the GitHub repository. (You should always pull before you push). Print hub status. Print User B status.  
    
14. User A change something in bloodgroup.c. Print local status User A.  
    
15. User A adds its changes and commits at once. Print local status User A.
16. User A pulls repository from GitHub
17. User A pushes changes to GitHub repository. Print status User A.
18. User B pulls repository from GitHub. All 3 repositories have same copy now.
19. Then User A changes bloodgroup.c at two places and User B changes same bloodgroup.c at two places (make sure they change same general area of the code or in the next step there may be no merge conflict visible ) ϑ . Both add and commit changes. Print status of User A and User B.
20. User A pulls from GitHub and then pushes to GitHub. User B then pulls and pushes to GitHub, but there’s conflict. (There will be a conflict if user A and B both change bloodgroup.c at about the same lines). Resolve this merge conflict for User B and commit change. Show each step.
21. Then User A pulls the changes. Display the contents of bloodgroup.c for both users to demonstrate that they are now the same.