**Homework 1**

1. Fetch?

We press fetch when we need to check for any updates. Then we can see if there are any remaining pulls.

1. Pull?

We pull when there are any updates in GitHub. This way we get the updated code into our local repository.

1. Push?

When we push a committed piece of code, we are actually pushing our code into the online repository.

1. Commit?

When we are committing a code, we are saving that modified code into our local repository.

1. Differences between commit and push?

Commit saves the modified piece of code into the local repository whereas push modifies the code in the online repository.

1. How to create merge conflicts?

In order to create a merge conflict, one person who is working on the same line of code must push the code before the other person pushes. Then, when the other person pushes that code, he will face a merge conflict.

1. How to resolve merge conflict?

The person who is facing the merge conflict must remove the head and the < and > symbols and then somehow concatenate the code and must commit the code. Then he must pull and then push.

1. Commit message why they are important?

When there are several people who are working on the same code, the commit message must briefly describe what the changes made in the previous commits are so that the other contributers will know what changes were made.

1. Why is it a good idea to pull before you push?

Everytime we push, we need to pull in order to see if there is an already updated code in the same line that we are updating. Therefore, this will prevent a merge conflict.

1. Why is it a good idea to commit before you pull?

We do this in order to avoid merge conflicts