**Qt Creator and Git (in existing project)**

1. Establish connection with GitHub how to do that is written below (at the bottom of this document)
2. You need to create new repository in opened project. In Qt go to Tools -> Git -> Create resository… -> indicate the folder with your project. If you have a project in folder *C:\Users\a\Desktop\Dokumenty Wiktora\V rok\Computer Graphics\Assignment\AssignmentBednarek*

When pop-up window shows you should be inside *AssignmentBednarek* folder I that case. You can go up in pop-up window in to higher folder and indicate *AssignmentBednarek* folder. Or if you click ok inside that folder it should be also ok. I don’t know if you need to do that step.

1. Tools -> Git -> Local repository -> Commit-> make new local commit and name it
2. Tools -> Git -> Remote repository -> Manage Remotes -> under the *Name* section type : *origin*

under *URL* type your project URL for instance: [*https://github.com/WBednarek/Computer\_Graphics.git*](https://github.com/WBednarek/Computer_Graphics.git)

**Be careful! Use https links not ssh!**

1. Tools -> Git -> Remote repository ->Push -> you will push your project on GitHub
2. If it still doesn’t work do following:
   1. Open Git bash -> go to your project folder -> type *git status* to be sure if you are inside folder with initialised git
   2. Put those commands:

git remote add origin [git@github.com:WBednarek/Computer\_Graphics.git](mailto:git@github.com:WBednarek/Computer_Graphics.git)

or

git push --set-upstream origin master

**Establishing Github SSH connection on Windows**

1. Run Git bash

1. Create ssh key using following command: *ssh-keygen -t rsa*

1. Save it in location you want (default is /c/Users/Domowy/.ssh) and it is recommended to create strong passphrase

1. Paste your public key (with .pub extension) to github website View profile and more -> Settings -> SSH and GPG keys

1. Check if your key exists: *ls -al ~/.ssh*

1. Add your SSH key to the ssh-agent using *eval* command as below:

•  **if you are using Git Bash**, turn on ssh-agent:

# start the ssh-agent in the background

*eval "$(ssh-agent -s)"*

Agent pid 59566

•  **If you are using another terminal prompt**, such as [Git for Windows](https://git-for-windows.github.io/), turn on ssh-agent:

# start the ssh-agent in the background

*eval $(ssh-agent -s)*

Agent pid 59566

1. Add your SSH key to the ssh-agent. If you used an existing SSH key rather than generating a new SSH key, you'll need to replace id\_rsa in the command with the name of your existing private key file. Use command:

*ssh-add ~/.ssh/id\_rsa*

1. Check your ssh connection: *ssh -T* [*git@github.com*](mailto:git@github.com)*. Check whenever you have your username in output prompt*