

# 1 Introduction

Welcome to the *Computational Solid Mechanics class 17/18!* We've created this document to make the setup and coding aspect of the project assignments as easy as possible. We want you to be able to focus on the theoretical side of things while having some nice computational results. This document includes:

- Instructions for setting up your computer with appropriate software
- How to log onto the computation solid mechanics server (CSM-server)

In the interest of readability all text entered into the terminal adopts the following form

```
>stuff typed into terminal
```

Many thanks to Ishan Tembhekar, for providing us with the project script for the Ae106 Caltech course.

## 2 Username, Password & Change of Password

In the following sections, we will introduce you to logging onto the server and doing all kind of crazy things on there. During your time on the server, you will be logged in under your username. Upon connecting to the server you will furthermore be prompted to enter your password. By default, these are given as follows:

- **Username:** All letters in lowercase: **first letter of your first name** followed without any space or other letter/digit by the **entire last name**. This being said, Max Mustermann would get the account *mmustermann*
- **Password:** All letters in lowercase: your **entire first name** followed by the last three digits of your Legi-No. This being said, if Max Mustermann has the Legi-No. 17-123-456, then his default password would be *max456*.

No matter how you first log onto the server (which is explained in the following sections), please change your password the very first time you log onto the server. This is accomplished via

```
>passwd
```

You will first be prompted to enter your current password, followed by the choice of your new password and finally confirmation of your new password. This is for your own security. Cheers!

## 3 Coding environment - editing files

Once you are on the CSM-server, you might want to start editing files, run simulations and ultimately be able to produce visualizations. There are essentially two approaches that you can take

1. Option 1: Edit files on your own computer using your favorite development environment (from gedit, vi, Emacs over Notepad++ to Microsoft Visual Basic++ - whatever floats your boat) and then transfer these files to the CSM-server once you are happy with your latest edit. We highly suggest this procedure for people that do not bring a great expertise from previous coding classes.
2. Option 2: Edit files on the CSM-server using editors that are available on this virtual machine. This option saves the pain of transferring files around every time you make a small change, but it means you have to edit using editors available on the server.

3. Option 1.5: Certain development environments, such as Notepad++ allow to connect to the CSM Server. When editing files on the editor, this automatically is being uploaded onto the server, thus to some extent combining the virtues of Option 1 and 2. Details on how to use/make the best out of Notepad++ can be found in Sec. 8.

If you choose the 2nd method, then there are essentially 3 main editors that the CSM-server can provide you.

- **gedit** : *Recommended for beginners in programming*

To start using gedit on the CSM-server, just start up gedit by entering the following into the console

```
>gedit &
```

The & is important because that makes sure that you can continue to use your terminal for other commands while simultaneously using gedit. Once you open gedit, you can search for the file that you want to edit using the GUI and after you are done editing, save and close gedit.

- **vi** : *Recommended for really cool people*

To enter one side of the holy war and edit a file using vi, enter

```
>vi someFilename.cc
```

you can find a list of vi commands from [http://www.atmos.albany.edu/daes/atmclasses/atm350/vi\\_cheat\\_sheet.pdf](http://www.atmos.albany.edu/daes/atmclasses/atm350/vi_cheat_sheet.pdf). If you want to exit vi, enter “:q” and then hit enter. If it’s mad at you about unsaved changes, tell it that you mean it by entering “q!” and then hit enter.

- **Emacs** : *Recommended for really cool people who don’t like vi* To enter the other side of the holy war and edit a file using emacs, enter

```
>emacs &
```

you can find a list of emacs commands from <http://www.mcs.sdsmt.edu/lpyeatt/courses/EmacsCheatSheet.pdf>.

## 4 Working on the CSM-server

### 4.1 Accessing and Logging in

#### 4.1.1 Using Windows

**Things that you do the FIRST time:** In order to remotely log onto the server we use the so-called *ssh*-protocol. This is rather easy on Unix-based system as Sec. 4.1.2 will show. If you are on a Windows machine though and wish to access the server, you need a terminal, which permits to enter said *ssh* connection. A popular and simple choice providing the most important features is **PuTTY**.

1. Download PuTTY, e.g. from <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html> and install it - this should be straightforward, but do let us know if you have any issues.
2. Open *putty.exe* (should either comfortably lie on your Desktop or in *Program Files* or *Program Files x86*, which should open a window similar to the one illustrated in Fig. 1.

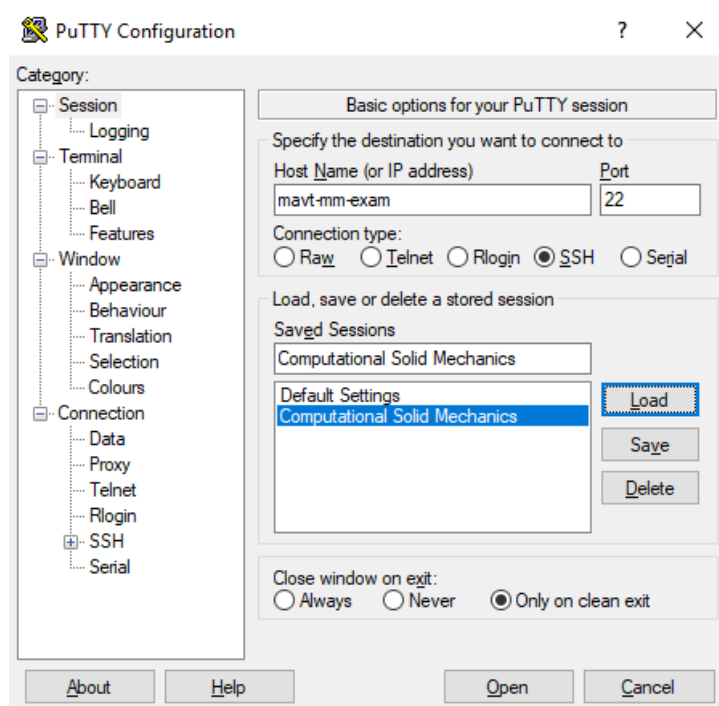


Figure 1

3. In **Host Name (or IP address)**, enter **yourUsername@mavt-mm-exam**, where the username was defined in Sec. 2.
4. In **Port** enter the number **22** (which is the standard port for SSH).
5. Check that the **Connection type** is set to **SSH**
6. In the *Category* section, follow the radio button *Connection* and then *SSH* and finally *X11*. Please tick the box **Enable X11 forwarding**. X11 forwarding allows to open graphical user interface (GUI) providing software. On Windows, the usage of such further requires a server providing the X11 capabilities, such as for example Xming (see Sec. 9), but one step after the other.
7. Going back a few steps to the radio button *Session* in the *Category box*, enter a name in the **Saved Sessions** box, such as for example *CSM* and press **Save**. This provides the comfort of simply pressing **Load** for future attempts to log in.

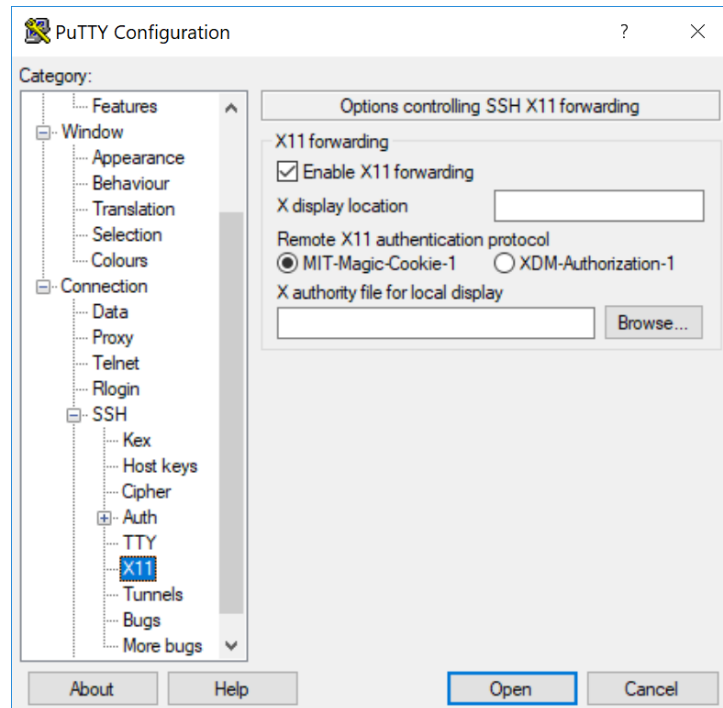


Figure 2

8. You're all set. Press the **Open** button. This should open a terminal and ask for your password. The username should not be asked for again, since we've already passed it in the **Host Name** section. The initial password is given in Sec. 2 and (unless the system asks you to change it anyways), you should definitely change said password straight after logging in as explained in Sec. 2.

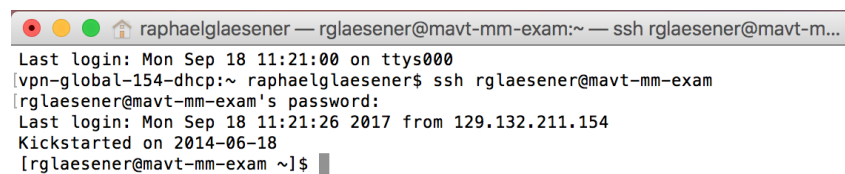
To exit the terminal just type **exit** in the terminal.

### 4.1.2 Using Linux or OS X

If you are using Linux or OS X - congratulations! For you, it should be extremely easy to log in.

1. Open a new terminal
2. Type in: **ssh yourUsername@mavt-mm-exam**
3. When prompted for your password, enter the default one given in Sec. 2.
4. You're logged in on the CSM-server! Please, asap change your password as explained in Sec. 2.

To exit your session on CSM, just type **exit** into the terminal. A sample session is shown in Figure 3



```
raphaelglaesener — rglaesener@mavt-mm-exam:~ — ssh rglaesener@mavt-m...
Last login: Mon Sep 18 11:21:00 on ttys000
[vpn-global-154-dhcp:~ raphaelglaesener$ ssh rglaesener@mavt-mm-exam ]
[rglaesener@mavt-mm-exam's password: ]
Last login: Mon Sep 18 11:21:26 2017 from 129.132.211.154
Kickstarted on 2014-06-18
[rglaesener@mavt-mm-exam ~]$
```

Figure 3

## 5 Compiling and Running

Once we have edited our file and feel that we are ready to see what results it gives us, we need to follow two important steps. First we **compile** the file to convert the human readable C++ to a binary. Following that, we **run** the executable to obtain our results. So how exactly do we do it?

1. Save your edited files and make sure you saved the right ones. Sounds stupid, but you won't believe how many times something did not show up, simply because we edited a file with the same name in another folder.
2. On your terminal, reach the directory where the executable is located.

```
> cd path/to/executable/location
```

which in our case is generally

```
> cd /home/username/HW1
```

3. Once we reach there, we type **make** followed by the name of the file we were editing *without the .cc* and press ENTER.

```
> make NameOfExecutable
```

Alternatively, typing

```
> make
```

would compile all .cc files listed under \$TARGETS in the Makefile.

4. Wait for the compilation to be over - in a) the case of errors, edit the files and re-try or in b) the case of a successful compilation, **run** the executable that has just been formed by typing the following in the terminal.

```
> ./NameOfExecutable
```

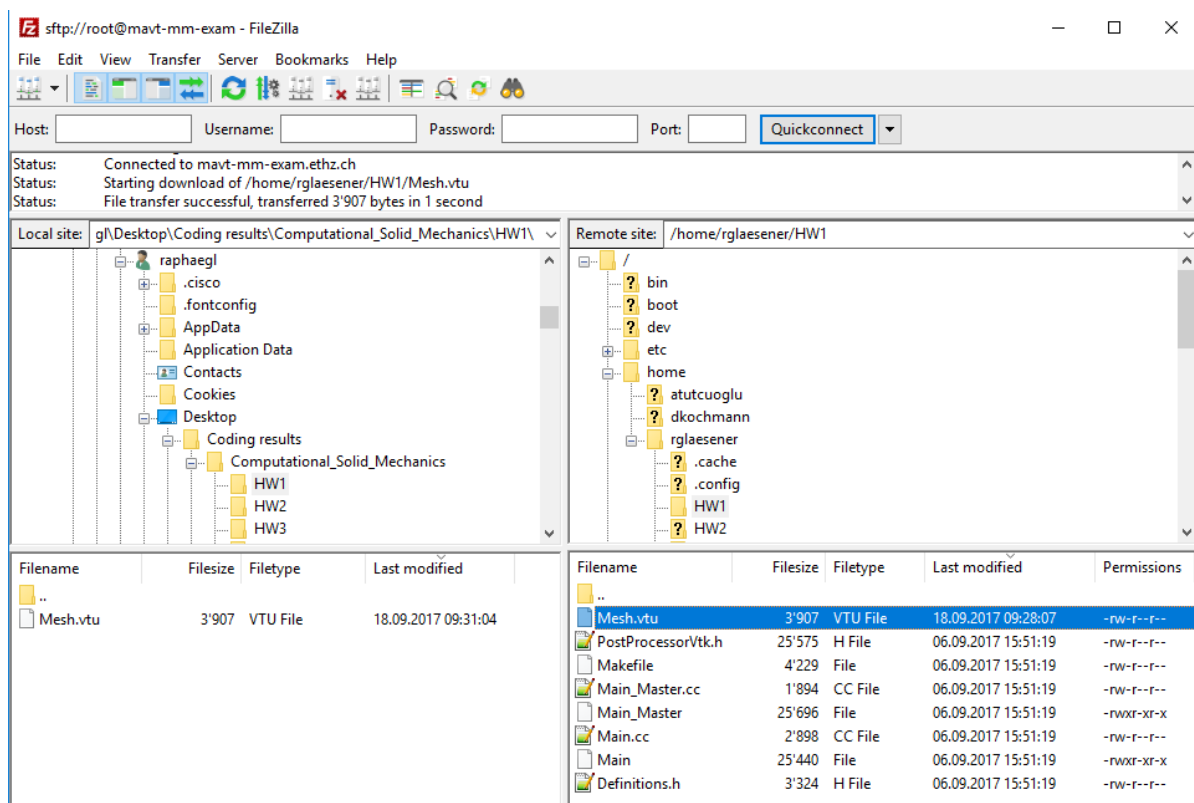
And that's it! We have to keep repeating this process for every file that we work on. Most of the times, when we are constructing the code, we get errors when we compile. So, the general approach towards successful compiling and running is as follows:

1. **Edit** your file
2. **compile** it by executing **make NameOfExecutable**
3. Get pages full of errors (a warm welcome to the nitro-templated Eigen library)
4. Go back to step 1 and edit the file to take care of those errors (repeat steps 1 and 2 for as long as necessary)
5. Finally, write an error free version of the file and compile it without errors (Note: No compilation errors does not mean that everything's working. It just means that the code is free of syntax errors. Our logic can still be wrong)
6. Then we **run** it by executing **./filename**
7. See if the results are what we expected.
8. If yes, celebrate and go on vacation! If no (which happens most of the time), go back to step 1 and repeat

## 6 FileZilla (Windows & Linux & MAC OS)

There are many options for transferring between a server and your computer and we like FileZilla the best. You'll need it to pull data or plots from the CSM-server to your computer. You can download FileZilla from <http://filezilla-project.org>. After installation the first step is to connect FileZilla to the server. In case you decide to use Notepad++ on your Windows, you can skip this Section as Notepad++ has an AddOn for transferring files from the server to your laptop (see Sec. 8).

- **Host:** sftp://yourUsername@mavt-mm-exam, where yourUsername is defined in Sec. 2
- **Username & Password:** as explained in Sec. 2
- **Port:** 22



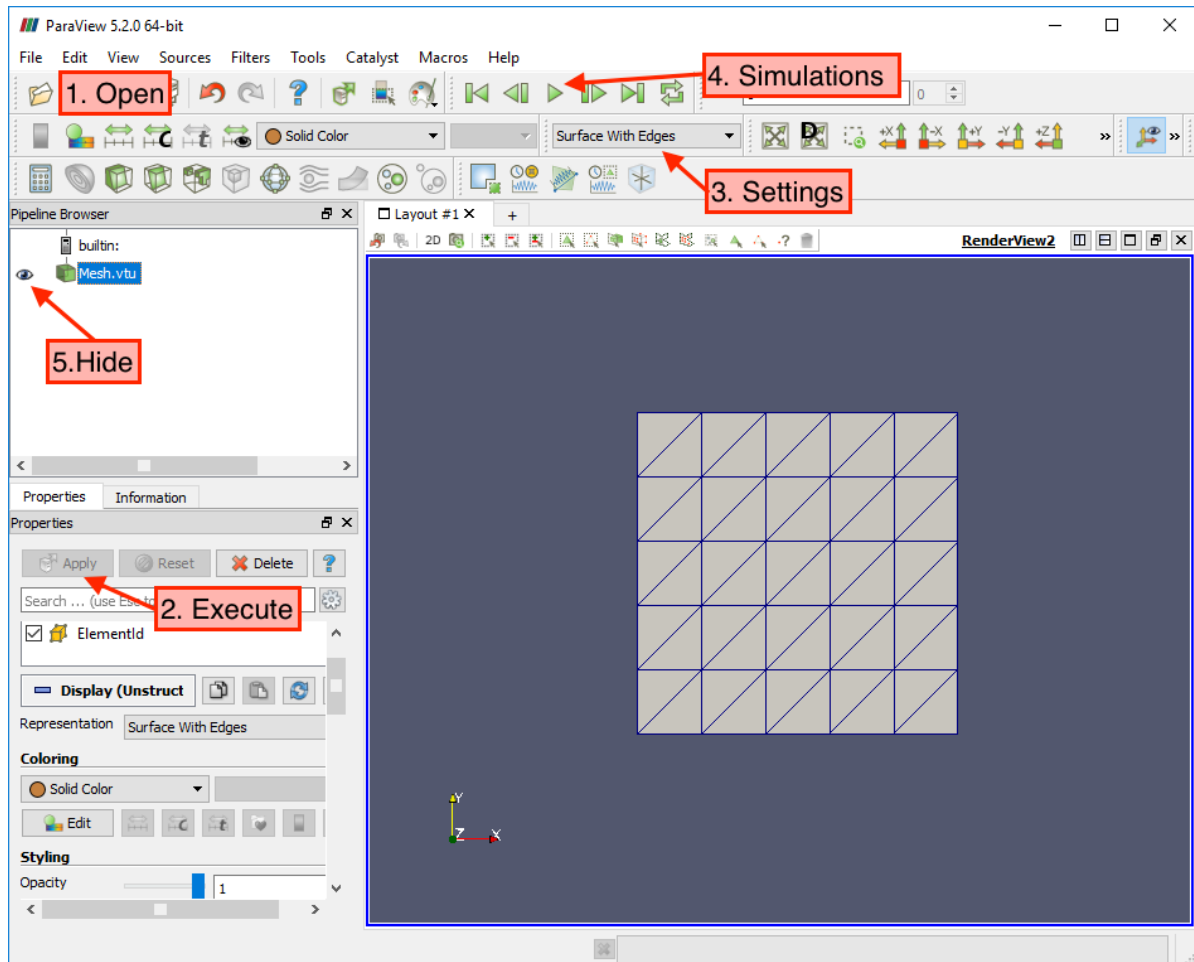
The **right side** of the screen represents the directory on the CSM-server while the **left side** depicts the directory on your own computer. To access the **project assignments** go to `home/username/HW1`. In here you will find **Main.cc** (in latter assignments this might include other .cc's, too) as well as a selection of libraries (**Definitions.h** & **PostProcessorVtk.h**). Files can be transferred between your computer and the server simply by **drag and drop**: When working on the project assignment, **pull** the code on your computer, **make changes** to it using any coding environment (depending on the OS you are using), save it, and then **push it** back onto the server.

The very same procedure has to be conducted for the results. By default the results are saved in the same folder as the executable. Of course you can also create your own output directory and save your results in there by making the appropriate changes in the code. To analyze these output files, drag and drop them onto your computer and open them using Paraview (see Sec. 7).

## 7 Paraview (Windows & Linux & MAC OS)

Within our code, we save data in binaries with extensions *.vtu* and *.vtk*. These output files are read by a visualization software called **ParaView**. You can download ParaView from <http://www.paraview.org/paraview/resources/software.php>. I **recommend downloading the older version ParaView 5.2.0** as it is the most stable one and has all the functionalities that we need (to be more precise, there may be issues with Intel Onboard GPUs with some of the latest versions).

To visualize your results, the following steps should be followed:



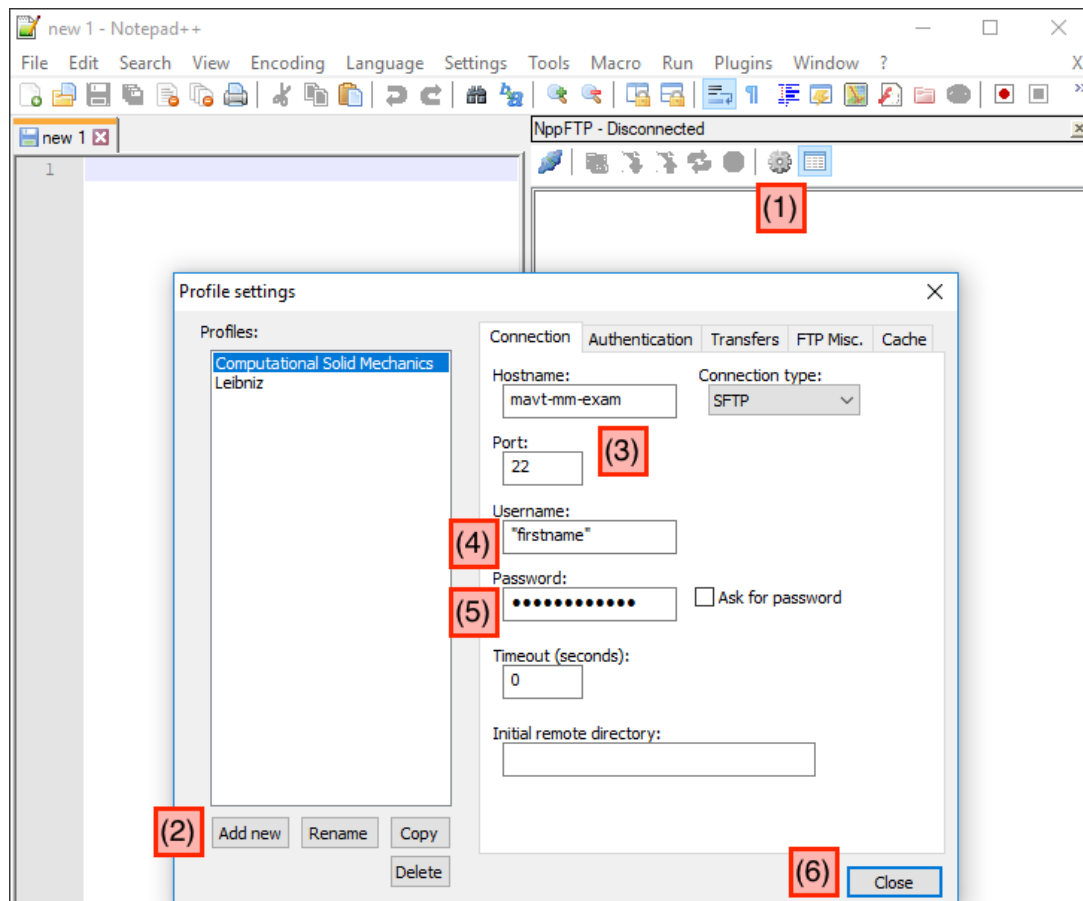
1. *Open* the file from the directory you have saved it in.
2. **Apply** to execute the visualization.
3. Change the *visualization settings*
4. If you have a simulation (starting HW5) you can sweep through the individual loadsteps using the *play button*.
5. You can *hide* the visualization using the *small eye-symbol*.



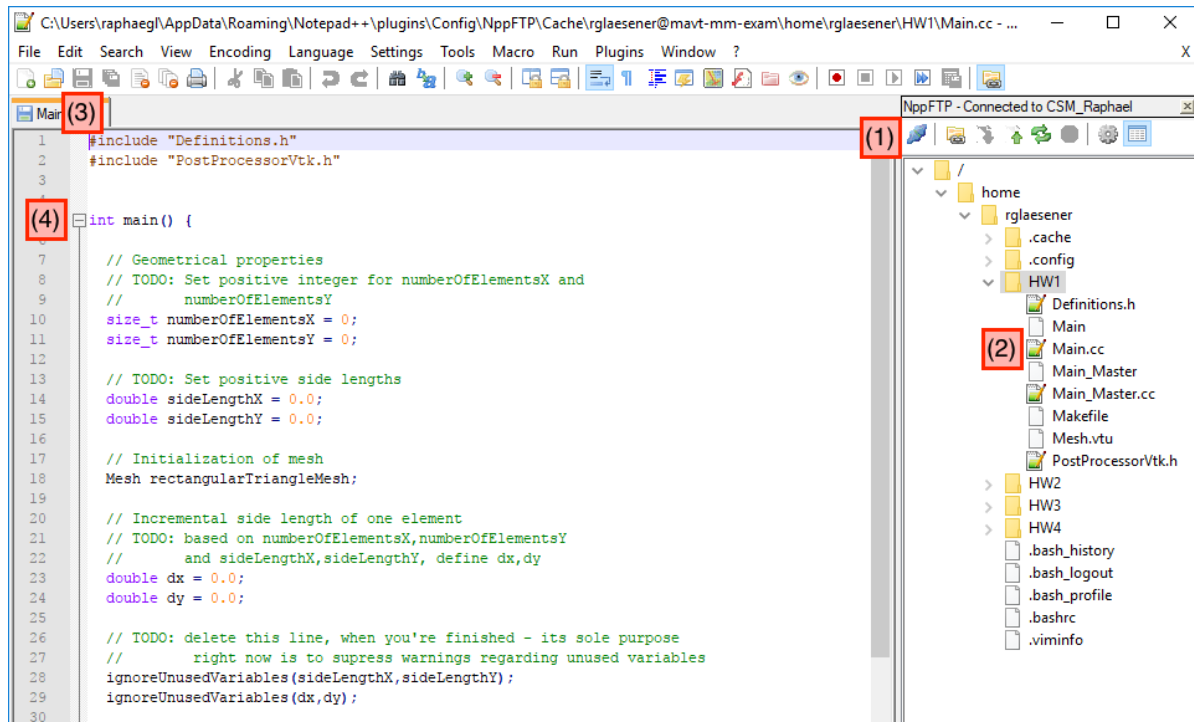
## 8 Notepad++ (Windows)

Notepad++ has the major advantage that it offers a coding environment including the ability to load plugins so as to allow for more advanced features. Prominent examples include a comparison tool (*Compare*) as well as an SFTP client (e.g. *NppFTP*) providing us with the option to connect to a server and hence essentially modify files directly on there. It can be downloaded from: <https://notepad-plus-plus.org/download/v7.5.1.html>. To connect to the CSM-server you need to follow these steps:

1. Click on the small gear in the top right
2. Add new server and give it an arbitrary name
3. The hostname is mavt-mm-exam, the type is SFTP and the port is 22
4. Enter your username as presented in Sec. 2.
5. Enter your password as defined in Sec. 2, unless already changed.
6. Close the window.



To work with Notepad++, you first need to connect to the CSM-server. Click on the connect button (1), choose the server and the directory will appear on the right side of the screen. Access the project assignment located in *home/username/HW1* and open the file you would like to edit (2). Save your changes on the server prior to compilation (3). A nice tool of Notepad++ is the option to collapse functions (4). This can be helpful when working on long code segments in that it grants a nice overview.



## 9 Xming

On Windows, the ability to open graphical user interfaces on separate windows other than the terminal not only requires `-X11` to be switched on, but furthermore a X11 display server. Xming is one of such solutions, comfortable not only because of the simple installation but also its compatibility with SSH connections established via PuTTY.

1. Download, e.g. from <https://sourceforge.net/projects/xming/files/latest/download>
2. Install.
3. Start Xming Server. It should appear in the background as illustrated in Fig. 4.

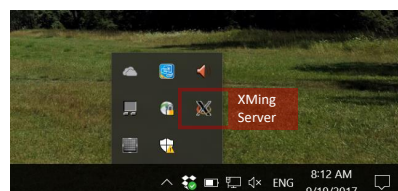


Figure 4: As soon as the Xming Server appears (and presuming everything has been properly installed), you're good to go for some serious *X11*.