**Case Study Full Steps**

**You may choose one of the following Case studies :**

1. You arranged an internship at a local pharmaceutical company with a team working on strategies for inhibiting venom binding to the enzyme acetylcholinesterase. This enzyme breaks down the neurotransmitter acetylcholine at the synaptic cleft (the space between the two nerve cells) allowing transmission of subsequent nerve impulses. The green mamba and other venomous snakes produce a deadly toxin, which is an inhibitor of acetylcholinesterase. You need to design a drug target to prevent the interaction between the toxin in snake venom and acetylcholinesterase. You may use the ray fish Acetylcholine Esterase with pdb **1ACJ.**
   1. Follow the steps for Molecular Docking with Autodock Vina and report the affinity of Tacrine to Acetylcholinesterase.
   2. From the “compounds used in projects”, find out more inhibitors other than Tacrine that succeeded to inhibit the Acetylcholinesterase.
2. Acetylcholinesterase and Butyrylcholinesterase are among the cholinesterase family that are nearly similar in active site. According to the inhibitors that affect their activity in this article 10.3390/ijms15069809, it is observed that both of them are inhibited by Tacrine, but the inhibition of Acetylcholinesterase by Tacrine (1ACJ) more than that of Butyrylcholinesterase (4BDS).
   1. Prove that the inhibition of Acetylcholinesterase by Tacrine more than that of Butyrylcholinesterase
   2. From the “compounds used in projects”, find out more inhibitors other than Tacrine that succeeded to inhibit the Butyrylcholinesterase.
3. Inflammation occurs when the body’s protective mechanisms are triggered in response to injury. The accumulation of white blood cells at the site of injury leads to pain, swelling, heat, redness, fever, and pain, which are together referred to as inflammation. Prostaglandin H synthase (PGHS) is a rate-limiting enzyme in the production of prostaglandins and thromboxane, which are important regulators of vascular function. Under normal physiological conditions, PGHS-dependent vasodilators (such as prostacyclin) modulate vascular tone. Ibuprofen is a non-selective inhibitor of an enzyme called cyclooxygenase (COX), which is required for the synthesis of prostaglandins via the arachidonic acid pathway. COX is needed to convert arachidonic acid to prostaglandin H2 (PGH2) in the body. PGH2 is then converted to prostaglandins. The inhibition of COX by ibuprofen, therefore, lowers the level of prostaglandins made by the body. Using the accession **1EGQ** of OVINE COX-1 COMPLEXED WITH IBUPROFEN , Would you be able to discover new inhibitors with affinity similar to or even better than Ibuprofen?
   1. Follow the same steps for Molecular Docking and report the affinity of IBUPROFEN to COX-1.
   2. From the “compounds used in projects”, find out more inhibitors other than IBUPROFEN that succeeded to inhibit the COX-1.
   3. Check another 3 **NSAIDs (Non steroid anti inflammatory drugs** from public databases (such as Pubchem) and report their affinity compared to IBUPROFEN.

#### Crystal structure of mixed-lineage kinase MLK1 complexed with compound 16 under the accession **3DTC**,

#### Try to follow along with the virtual screening steps to check if you can find more compounds with better affinities.

* 1. Try to get other 10 compounds from Zinc15 database using the gene encoding of the target protein and check their affinities to the target compared with the co crystallized ligand

**Deliverables**:

* A final conclusion of your results
* Visualization of the protein with the top 3 compounds of the best affinities ( from the provided library, drugs, or public databases). You may check the recorded video for visualization available on week 9.
* A zip file containing the final output file sorded according to the best affinity.

**Revision for the whole steps needed for docking**

Those steps are covering the steps from the very beginning of choosing the target protein of interest, meanwhile they are provided to help you through case study, so you may bypass step 2 and 3.

Credits for the detailed steps mentioned goes to our colleague: Ruwaa Ibrahim

# Preparing the working directory and environment.

* 1. Create a new working directory for the Case study in the **Home** directory. Name it by your enzyme class number, for example **EC1**.

This can be done via the following commands:

$ mkdir ~/EC1

* 1. Create the following empty directories (case sensitive):

**target, ligands, Log, out**

This can be done in the terminal via the following commands:

$ cd ~/EC1

$ mkdir target ligands out Log PDB\_files

* 1. Download the required Python scripts from Moodle.
     1. Two Python files on Moodle, inside the week 8, named “docking.py” and “log\_extraction.py”. They should be downloaded and placed inside the main project working directory either via GUI or by executing the following command after download.

$ mv ~/Downloads/docking.py ~/Downloads/log\_extraction.py ~/EC1/

* 1. Download the virtual screening library from Moodle.
     1. One compressed file named “Project-pdbqts.zip” on Moodle inside the week 9 tile. Download it, unzip the file, move the molecules from the created directory to our project directory inside the ligands. You can use the following commands for the same purpose.

$ mv ~/Downloads/Project-pdbqts.zip ~/EC1/

$ cd ~/EC1

$ unzip Project-pdbqts.zip -d ./ligands/

* 1. Install the required programs.
     1. **PyMOL**
        1. From the terminal, execute the following commands:

$ cd ~/EC1

$ wget https://pymol.org/installers/PyMOL-2.5.2\_293-Linux-x86\_64-py37.tar.bz2

$ tar xjf PyMOL-2.5.2\_293-Linux-x86\_64-py37.tar.bz2

* + - 1. To run PyMOL, run the following command

$ ./pymol/pymol

* + 1. **MGLTools**
       1. From the terminal, execute the following commands.

$ cd ~/EC1

$ wget http://mgltools.scripps.edu/downloads/downloads/tars/releases/REL1.5.6/mgltools\_Linux-x86\_64\_1.5.6\_Install

$ chmod a+x mgltools\_Linux-x86\_64\_1.5.6\_Install

$ ./mgltools\_Linux-x86\_64\_1.5.6\_Install

* + - 1. Click Next and follow the installation process
      2. To run MGLTools, execute the following command:

$ ~/MGLTools-1.5.6/bin/adt

* + 1. **AutoDockVina**
       1. From the terminal, execute the following commands:

$ cd ~/EC1

$ wget <http://vina.scripps.edu/download/autodock_vina_1_1_2_linux_x86.tgz>

$ tar xzvf autodock\_vina\_1\_1\_2\_linux\_x86.tgz

# Selecting a protein with no 3D structure on PDB.

* 1. Open Uniprot (<http://uniprot.org/uniprot/>) and click **search**.
  2. On the top of the results table, click on the edit sign.
  3. From the tab **3D structure**, select **PDB** and click save at the end/top of the page.
  4. In the search bar, write the following search term, changing the EC number to the enzyme class number of your project:

**ec:1.-.-.- NOT database:(type:pdb) AND reviewed:yes**

Alternatively, click on Advanced below the search bar and add the keywords.

* 1. (Optional) Add organism filter.
  2. Select a protein as needed. Copy its UniProt ID, for instance: Q8TDQ7.

# Perform Homology Modeling using SWISS-MODEL

* 1. Go to SWISS-MODEL (<https://swissmodel.expasy.org/>) and click **Start Modeling**.
  2. Select the input type **Sequence(s)**.
  3. Paste your protein UniProt ID in the search space and click **validate**.
  4. After the sequence has been imported from UniProt DB, click **Search For Templates**. This will take some time.
  5. From the results table, select a template based on the criteria discussed in lab 4. Make sure that you template has ligands.
  6. Expand the selected template, from Target Prediction select **monomer** instead of homo-x-mer then click **Build Model**. Note that this is not applicable in the case of hetero-x-mer.
  7. Go to the Models tab in the results page and explore your model. Apply the criteria discussed in lab 4.
  8. Expand the description part (by clicking the second small arrow in the model). Make sure that your model has a ligand. If not, please go back to the templates and select another template.
  9. Download the model by clicking on the **Model number** then select **PDB format**. Move it to the project directory (EC1) either via the GUI or by the following command.

$ mv ~/Downloads/model\_01.pdb ~/EC1/PDB\_files

# Prepare your files for docking/virtual screening.

* 1. Extract the chain and the co-ligand using PyMOL.
     1. Run PyMOL and open the downloaded model (File → open → browse to ~/EC1/PDB\_files and select the model.pdb file).
     2. Display the sequence of the protein using GUI through (Display → Sequence). The top line (in white) shows the model name, chain name (A/B/C/D), and amino acids’ numbers.
     3. Extract one chain (if it’s not a monomer) by the following command→

> extract chain\_A, chain A

* + 1. Note that the co-ligand of the chain is written with the same color as the chain. Extract that co-ligand by the following command:

> extract ligand, resn [the name of the ligand from the end of the chain sequence]

* + 1. Add hydrogen to all using the following command.

> h\_add all

* + 1. Save the selected chain and ligand -one at a time - using the GUI:

File → Export Molecule → select the chain from the dropdown menu → click save → name it chain\_A.pdb → browse to ~/EC1/PDB\_files → click save.

File → Export Molecule → select the ligand from the dropdown menu → click save → name it co-ligand\_A.pdb → browse to ~/EC1/PDB\_files → click save.

* 1. Determining the Grid\_box dimensions and converting the the protein to PDBQT format.
     1. Run MGLTools (ADT)
     2. Open the protein chain and co-ligand (if available) via the following steps.

File → Read Molecule → select your molecule (~/EC1/PDB\_files/chain\_A.pdb).

(If available) Ligand → input → open → select your co-ligand (~/EC1/PDB\_files/co-ligand.pdb).

* + 1. Display the ligand as atomic sphere model by clicking on the third circle in front of the co-ligand.
    2. Determine the Gridbox dimension
       1. Grid → Macromolecule → choose → chain\_A. This will ask you to save the protein in PDBQT format. Click OK. Browse to ~/EC1/target/ and save it there.
       2. Grid → Girdbox. Make the spacing 1 Å. Change box dimensions and position to cover the whole ligand. If no ligand is available, make the box cover the whole protein.
       3. In the gridbox window, click File → close saving current.
       4. Grid → gridbox → File → output grid dimensions file → name it “grid\_box.txt” and save it in the main working directory (~/EC1/).

# Perform Virtual Screening

* 1. Run the docking script

$ python3 docking.py

* 1. Run the log extraction script

$ python3 log\_extraction.py

# Selecting the Best Hits

* 1. Open the “Output.csv” in spreadsheets (Excel for instance of LibreOffice Calc) and sort the rows based on column B ascendingly. Save the modification. Now the best compounds are seen on the top of the table.

# Visualizing using PyMOL.

* 1. Open the protein and the best molecules from the output directory in PyMOL for visualization. Using the following code to visualize hydrogen contacts (prot = protein, lig=logand) [from [PyMOL wiki](https://pymolwiki.org/index.php/Displaying_Biochemical_Properties#Hydrogen_bonds_and_Polar_Contacts)]

> h\_add all

> select don, (elem n,o and (neighbor hydro))

> select acc, (elem o or (elem n and not (neighbor hydro)))

> dist HBA, (**lig** and acc),(**prot** and don), 3.2

> dist HBD, (**lig** and don),(**prot** and acc), 3.2

> delete don

> delete acc

> hide (hydro)

> hide labels,HBA

> hide labels,HBD

**Extra Steps from previous lab troubleshooting**

1. To convert the co-ligand to PDBQT format from MGLTolls, Ligand → output → save as PDBQT → browse to the desired location and name it chain\_A\_co-ligand.pdbqt
2. Use OpenBabel to convert SDF file downloaded from ZINC15 to PDBQT.
   1. Download OpenBabel: From the terminal, execute the following command:

$ sudo apt install openbabel

* 1. Convert the SDF downloaded from ZINC15 to PDBQT format

$ babel for-sale.sdf output.pdbqt -m