Exercise 6: Docking.

We propose the following problem: we have the structures of two proteins. We know that these two proteins interact, but we don't know how does this interaction look like nor the regions of the proteins that are interacting. To find which is the conformation of this interaction we can use docking programs. Besides, we can rebuild the interaction between two proteins using protein superimposition and the structures of homologous interacting proteins, as we learned in practical 3.

Theoretical concepts:

Docking: Docking programs explore the conformational space of the interaction between two proteins. Then, they obtain a set of possible conformations for that interaction that are ranked according to scores. These different conformations are also called docking poses. There are different algorithms to obtain the docking poses. These may use a systematic search, molecular simulations or local shape features. Besides, there are different scoring functions to rank the docking poses. These can be divided in two main groups: those that are based on chemical and physical properties, and those that are knowledge based. If you remember from other practices, statistical potentials were also knowledge based scores. Actually, some of the scoring functions used to rank docking poses are statistical potentials that have been designed to evaluate protein-protein interactions.

There are mainly two types of docking algorithms:

- Rigid body docking: it obtains docking poses by translating and rotating molecular structures as rigid bodies. This means that no structural changes are introduced in the molecules.

- Flexible docking: it works like the rigid body docking, but it allows a certain degree of flexibility in the molecules. Therefore, structural changes are introduced in the molecules.

Keep in mind that docking can be applied to other scenarios than protein-protein interactions. It can be used to study the interactions between other types of molecules. For example, docking approaches are also used to study protein-drug interactions or protein-DNA interactions.

When working with docking programs we have to define the receptor and the ligand. The receptor is the biggest molecule, while the ligand is the smallest. In protein-protein docking the receptor will be the biggest of the two proteins. In protein-drug docking the protein will be the receptor and the drug will be the ligand.

During the process of docking, the receptor remains immobile while the ligand moves around it, testing all their possible interacting conformations. This exploration of the conformational space can be more or less exhaustive, depending on how many conformations you test. The more conformations you test, the higher the computational cost will be. Each one of these conformations is called a docking pose, and can be represented by a pdb file.

During this tutorial we will learn how to work with the zdock web server (https://zdock.umassmed.edu/).

Tutorial:

Step 1: Ordinary docking

In this tutorial we will work with the interaction between two proteins: the RhoGAP GTPase and the RAS related protein RAB1A. The interaction between these two proteins is fundamental for proper signal transduction within cells. Here, RhoGAP is the biggest protein, so it will be our receptor, while RAB1A will be our ligand. The PDB entries for these structures are 1F7C_A for RhoGAP and 4FMD_F for RAB1A.

The results of this docking execution are already in your directory for today's practical. However, you can do the submission on your own by submitting your pdb structures into the zdock web server. Go to and upload the pdb structures for RhoGAP and RAB1A. Once you submit your job, this process can take several minutes to finish. So, go to this page where you can see the output zdock page for the same proteins you submitted: https://zdock.umassmed.edu/results/1f03791af7/.

Zdock generates 2000 different docking poses. All these poses are scored using a scoring function that uses shape complementarity, electrostatics and statistical potentials. From the 2000 poses, zdock gives you the best 10, ranked according with their scores. Also, it allows you to download the results for the 2000 poses as a text file.

The text file that contains the results the 2000 poses is structured into 7 columns:

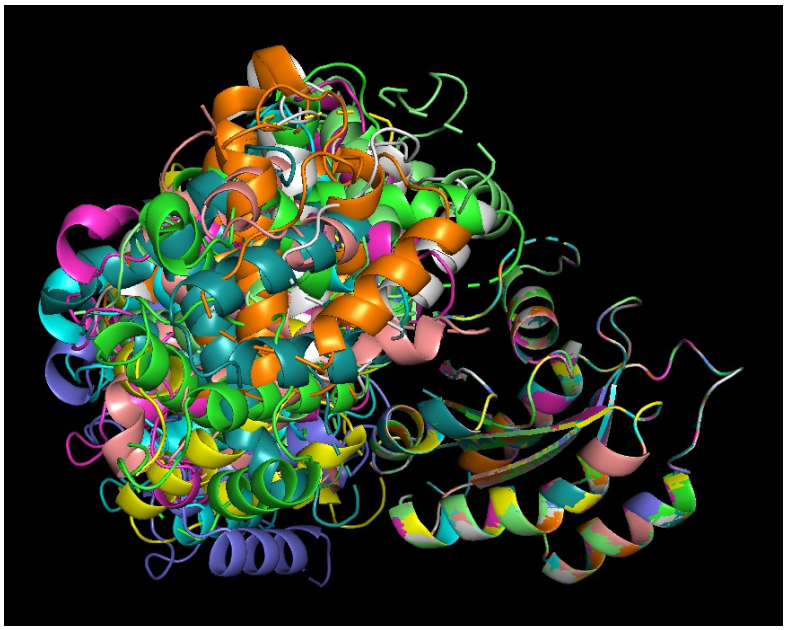
- Columns 1 to 3: They contain the rotation that should be applied to the ligand to achieve the conformation of the pose. There are 3 rotation values in radians for the 3 axis of 3D space.
- Columns 4 to 6: They contain the translation vector that should be applied to the ligand to achieve the conformation of the pose. There are 3 values for the 3 coordinates of the translation vector in the 3D space.
- Column 7: Contains the ZDOCK scoring function. The higher the best.

Let's start by comparing the different docking poses, this way we will see the diversity of poses given by a docking pose. Open pymol and load the 10 best scoring docking poses. Once you have done that, superimpose all the docking poses using the receptor as the reference structure:

```
super complex.2 and chain F, complex.1 and chain F
```

```
super complex.3 and chain F, complex.1 and chain F
```

See that we are using complex.1 as reference. We keep on making superimpositions until the 10 complexes are superimposed. See that the result is the receptor fixed, while the ligand is not, and we can see all the variability we have regarding the position of the ligand:



As you can see, the top scoring poses are very different among themselves. However, only one of these poses corresponds with the native state conformation for the interaction of the two proteins. Also, it is not sure that the best scoring pose is the native state one, neither that the native state pose is in the top 10, 50 or 100 best scoring poses. Here you can understand the main limitation of docking: finding the correct docking pose is like finding a needle in a haystack. Luckily, we have other resources besides docking scoring functions to identify docking poses.

Step 2: Compare the docking poses to experimental structures of the interaction. Using the 3did database.

So far we have been doing *ab initio* docking. As you have seen, it is hard to identify correct docking poses from all the results that docking programs produce. That is why the use of *ab initio* docking has some strong limitations.

Template guided docking on the other hand is way more reliable. This type of docking is what we did on practical 3, when we were using superimposition to reconstruct interactions between proteins. The main limitation of this approach is that there are few structures for protein-protein interactions. Therefore, if you want to model an interaction it is very likely that you cannot find templates for it, and then you have to use *ab initio* docking.

Another limitation of template based docking is that you have to find one PDB entry that is homologous to the two proteins that you are interested in. Then, you have to generate two sequence searches and look close to both files. This is very unproductive. Instead of that you can use 3did: https://3did.irbbarcelona.org/

3did is a database of interacting protein domains. You can search there for domains that you are interested in, and it will tell you what other domains interact with your query. Also, for each of these domain-domain interactions it will tell you if there is any PDB structure

representing this interaction. This can be an easy way to find templates for protein-protein interactions.

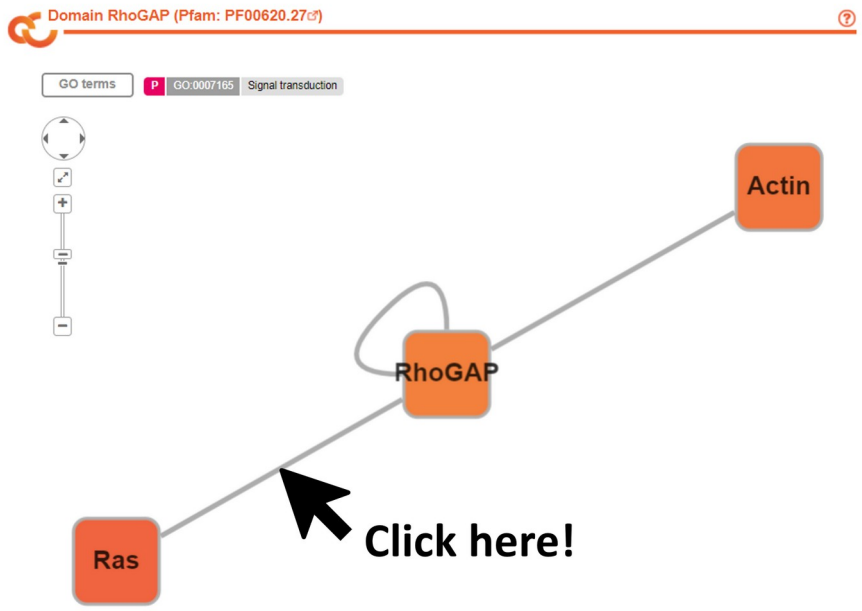
We are going to use 3did to find templates for the interaction we are studying. The first thing we need to know is what are the domains of our protein structures. To do this we will upload them to the cluster and execute hmmscan to see if they match with some HMM from PFAM. Remember to load the modules to execute the hmmer package.

```
hmmscan  /shared/databases/pfam-3/Pfam-A.hmm  ligandA.fa  >  ligandA.out
```

```
hmmscan /shared/databases/pfam-3/Pfam-A.hmm  receptorF.fa > receptorF.out
```

After this execution you know that the domains of your proteins are RhoGAP and RAS. Now you can go to the 3did database and search for one of these two domains, then check if it says that it interacts with the other.

Start by searching RhoGAP, you will see that this takes you to a page where you can visualize how other domains interact with RhoGAP. One of them is RAS. If you click on the node between RhoGAP and RAS you will arrive to a page describing this interaction.



Now let’s compare our docking poses with one of the structures containing the interaction between Ras and RhoGAP domains. We can select any of the structures containing the interaction between the two domains, but we can select the structure having the best database score: 5c2j.

This interaction has been found in the following PDB entries

PDB ID	Chain1	Residues	Chain2	Residues	Score	Z-score	Topology	3did	Visualization
5c2j	B	5-178	A	363-509	12.26	6.22981	0:0	View	Jmol
5c2k	A	7-180	A	225-371	11.41	5.30981	0:0	View	Jmol
5hpy	B	7-180	A	213-362	11.73	5.3741	0:0	View	Jmol
5hpy	F	7-180	A	213-362	3.82	3.66585	1:3	View	Jmol
5hpy	F	7-180	D	213-361	11.63	5.05579	0:0	View	Jmol
5irc	D	7-180	B	1262-1409	15.36	6.14173	0:0	View	Jmol
5irc	F	7-180	A	1262-1409	16.79	6.41745	0:0	View	Jmol
5irc	F	7-180	B	1262-1409	1.56	2.13388	3:1	View	Jmol
5jcp	A	7-180	A	920-1068	7.84	4.8677	0:0	View	Jmol

Start by loading this structure in pymol and superimpose the receptor of one of your structures on top of this structure:

```
fetch 5c2j
```

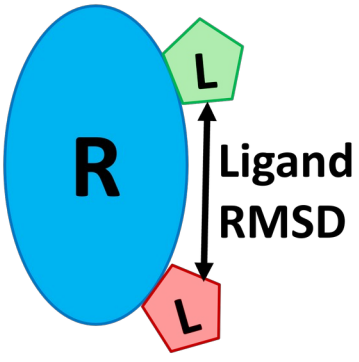
```
super complex.1 and chain F, 5c2j and chain B
```

Now we are superimposing the biggest chains in the interaction. However, the ligands are in very different location. If our docking pose was correct it would be exactly in the same position as the small chain of the 5c2j, but is not. Calculating the distance between the ligand of my docking pose and the ligand in the experimental structure would be a way to quantify the error in my docking prediction.

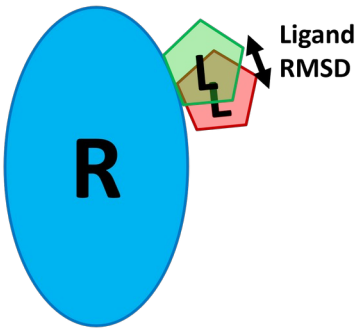
We can quantify this distance using the RMSD between the ligand in the experimental structure and the ligand in our docking pose.

- If the RMSD is very small this means that the docking pose is correct, because the ligand is located in a very similar position in comparison with the experimental structure.
- If the RMSD is very big this means that the docking pose is not correct, because the ligand is located in a very different place in comparison with the experimental structure.

If ligand RMSD is high: The docking pose is not correct



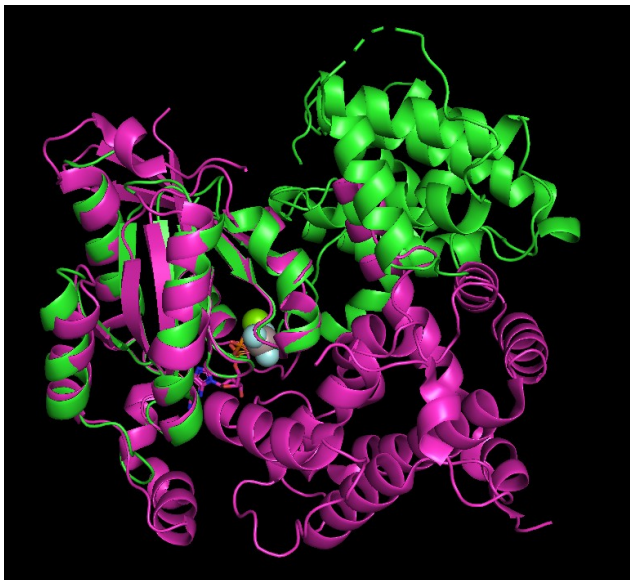
If ligand RMSD is low: The docking pose is correct



To measure the RMSD between two sets of atoms in their current positions we use the rms_cur command:

```
rms_cur complex.1 and chain A, 5c2j and chain A
```

In this case I get an **RMSD of 43.878** Angstroms. This is a lot, but it makes sense because the ligands are located in completely different locations:



In the previous image you can see the docking pose (in green) and the experimental structure (in purple). On the left side of the image you can see the two receptors superimposed, while the ligands are on the right. You can see that the ligands are in completely different locations.

The RMSD between ligands is a good measurement of the quality of each docking pose. We could ask ourselves, how well does this correlate with the scoring functions provided by ZDOCK? To answer this question I propose you the following exercise: choose 5 of the 10 docking poses we got from ZDOCK and calculate their corresponding ligand RMSD using as a reference the experimental structure we used before. Then compare their RMSD with their ZDOCK score, is there a correlation? To do that try to fill this table, you only have to repeat the procedure we just did but with 4 more docking poses:

Name of the pose	ZDOCK score	ZDOCK ranking	RMSD ligand	RMSD ranking
Complex.1	1206.272	1	43.878	?

Step 3: Comparing and analyzing protein interfaces

Another way to model protein-protein interactions is to use template guided docking (which is what we did at the end of practical 3). In this case, we can use the experimental structure as a template to reconstruct the interaction between the two proteins. We are going to do this by making superimposition:

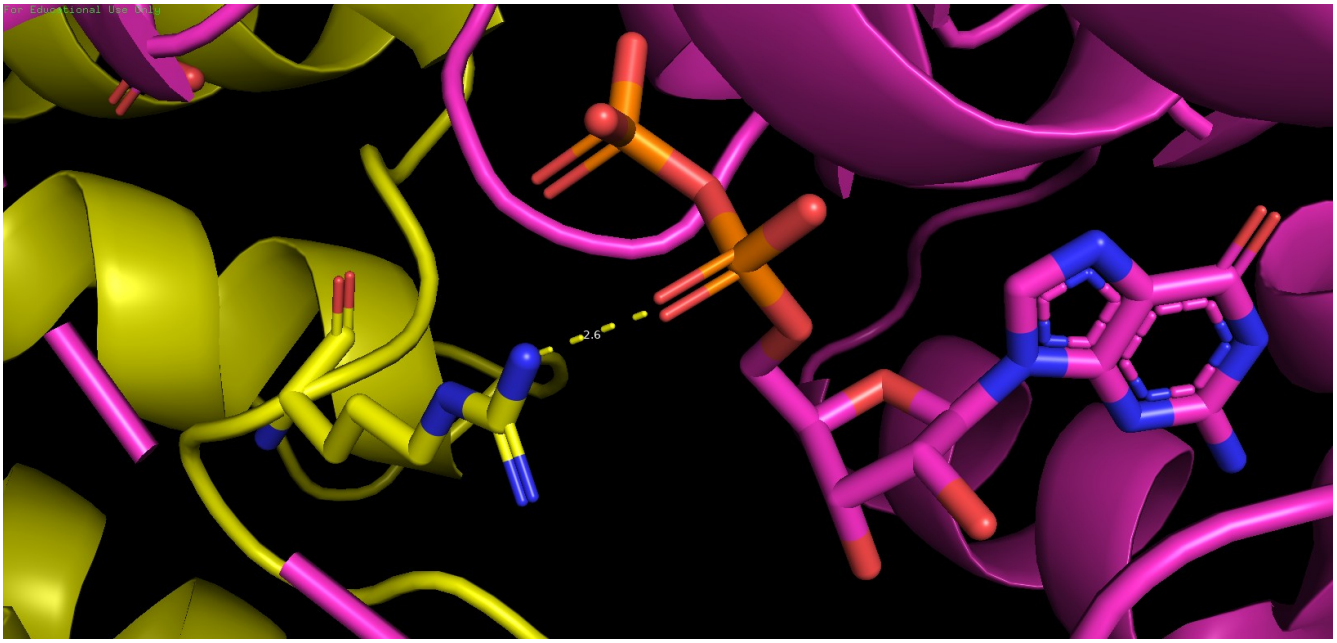
```
super complex.1 and chain A, 5c2j and chain A
```

```
super complex.1 and chain F, 5c2j and chain B
```

See that the subunits for the different poses are exactly the same, so the result of this superimposition would be the same for all docking poses.

Having done this, we are going to analyze the interface of our template structure and compare it to the one we just modeled. We will focus our analysis in identifying polar contacts such as hydrogen bonds or electrostatic interactions. We will do this by manual inspection of the structures. First, find polar contacts in your template interaction. Then, search for these same contacts in the modeled interaction. Can you find anything interesting?

Use the wizard > measurement option to measure distances in your pymol session. Remember that most polar contacts need to be in a distance within 2.5 to 3.5 Angstoms. By doing so, you can see interesting interactions such as an arginine of one subunit interacting with the phosphate groups of the GDP in the other subunit. This is an electrostatic interaction, since phosphate is negatively charged and arginine is positively charged.



Do you think that this interaction can take place in our model? Why? Can you see it? How would you reconstruct it?

Step 4: Finding interacting hot-spots to guide the docking

As you have seen, docking programs generate a huge ensemble of solutions, from which only a small fraction is correct. To improve this, most docking programs allow to restrict docking to some regions of the two proteins, forcing that certain amino acids should be included in the interface. This helps to reduce the number of wrong docking poses and helps to identify the correct docking poses.

We can use bioinformatical tools to find what residues are more likely to be interacting. These regions that are more likely to be involved in a protein-protein interaction are called hot-spots. There are several tools we can use to predict, in this practical we will learn how to use iFrag, one program developed by the group of Baldo. To access this method go to the following web page: https://sbi.upf.edu/web/index.php/research/servers/iFrag. Once you are inside the iFrag page, input the fasta sequences of the two proteins we are working with. You can get the fasta sequences from the P6_directory and click on submit.

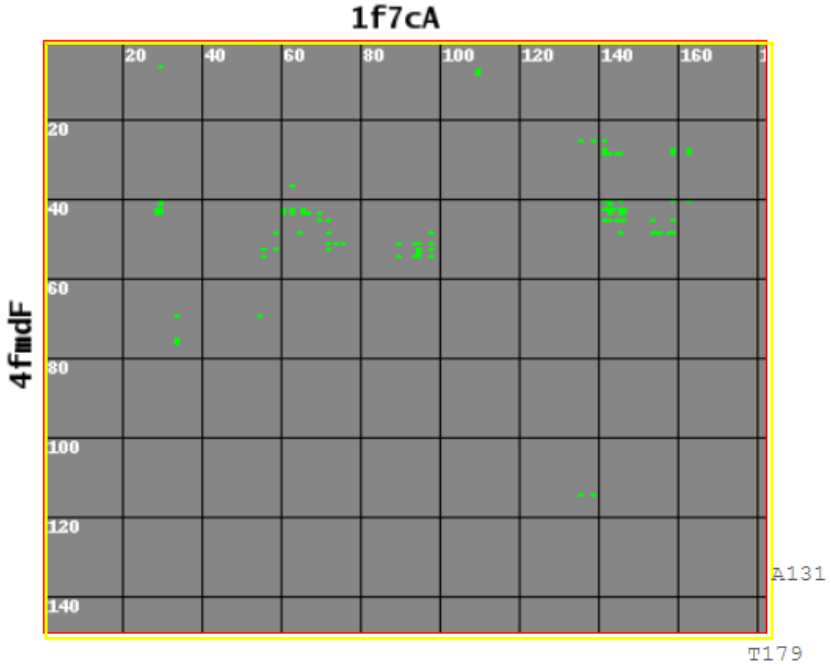
The iFrag server does several things:

- Checks in a database of experimentally determined protein-protein interactions for homologous sequences for the two input sequences. The objective is to find a pair of interacting proteins where each of them is homologous to one of the input proteins.

- Checks all the protein-protein interactions of the PDB and looks for homologous structures for the two input sequences. Again, the objective is to find interacting proteins that are homologous to the two input proteins. In this case, the results will be more precise, because iFrag will only show the regions that are interacting in the PDB structures.

- Checks the PFAM HMMs of the input proteins and looks for information of these two HMMs interacting. This option is not very precise because HMMs cover lots of amino acids, and here we are interested in the few residues that are involved in the interaction.

The results of iFrag are represented in hetmaps where the two sequences are shown in different axis. Residues that are predicted to interact are indicated with color. From all this information we will focus on the information provided by the PDB this corresponds with the next heatmap:



You can check the results for this iFrag execution at the next link: https://sbi.upf.edu/web/index.php/research/servers/iFrag?jobID=01121ba02bded48d1a8ca51ef3f604de

By checking this heatmap we can identify the following pairs of interacting amino acids:

Amino acids in RhoGAP	Amino acids in RAB1A
60-76: TITSALKTYLRMLPGP	26-31: TISTI
140-149: GVVFGPT	41-55: AGQERFRTITSST

Once you have identified the amino acids that are likely to interact, you can choose the residue selection option in the input page of ZDOCK and execute the program again. This will improve your docking poses, but still, you will have to discard most of them in order to find a successful docking pose.

Questions from the tutorial:

1) Try to execute zdock with a protein that you are working with in your projects. Then comment the results. Are all the top docking poses similar or different? What is the overall score for all the generated poses? Did you use some amino acid restrains to improve your results?

2) Try to find the protein you are working with in the 3DID database. What domains can your protein interact with? Do you have available structures for these interactions? Can you relate some of these interactions with the function of your protein?