# Report 1:

In this report we focused on getting setup for the project and afterwards implementing protein structure prediction on the csv file provided to us and getting our first results.

Our code is available in Github in the file DFold.ipynb

Our results are available in wandb.

Github: <https://github.com/daiyral/DFold>

Wandb: <https://api.wandb.ai/links/radiostars/ya2378wy>

## Setup

### 1. Installation

Repository: <https://github.com/evolutionaryscale/esm>

Install ESM3 using pip:

pip install esm

### 2. Environment

Python: 3.11

CUDA: 12.1

PyTorch: 2.3.1

## Pipeline Test

Before we begin, we wanted to install ESM and verify that the generation pipeline works. Specifically, we focused on the outer membrane protein from the PDB database.

### 1. Retrieve Protein Sequence

Retrieve the protein sequence for `1BXW`, chain `A`, from the PDB database.

We used the ESM3 class `ProteinChain` to retrieve the protein sequence using the method:

ProteinChain.from\_rcsb(1BXW)

Using py3Dmol, we visualized the protein.

A multicolored spiral structure

Description automatically generated with medium confidence

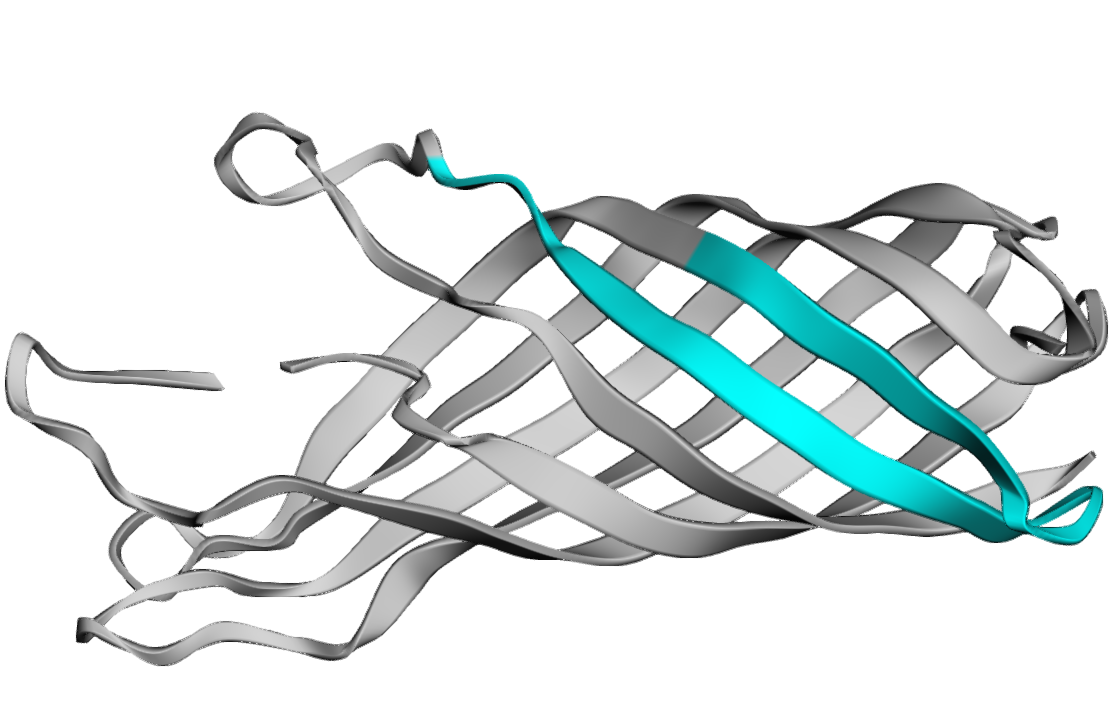
### 2. Selecting a Motif

We wanted to understand how to use the Protein class so we tried to select a distinct sequence of amino acids in the Protein.

The protein class in ESM3 can be used like a numpy array in the following way:

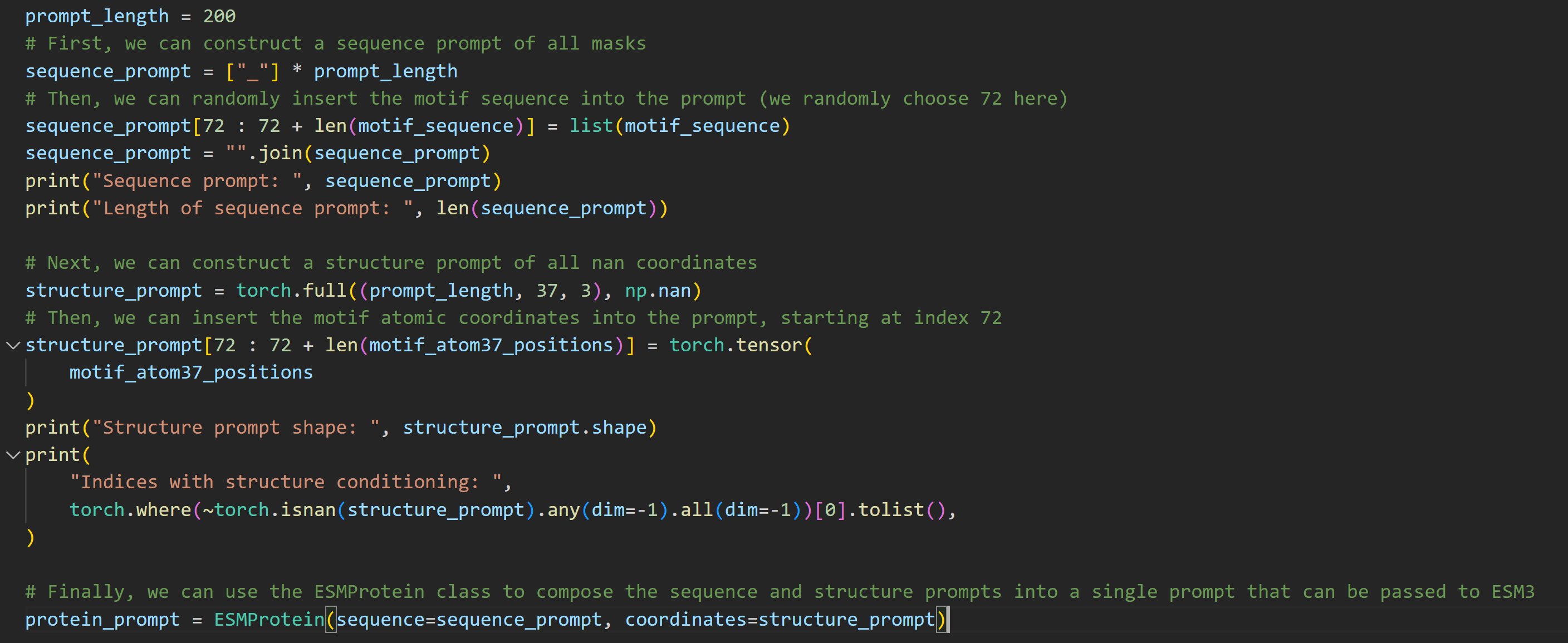
A black screen with text

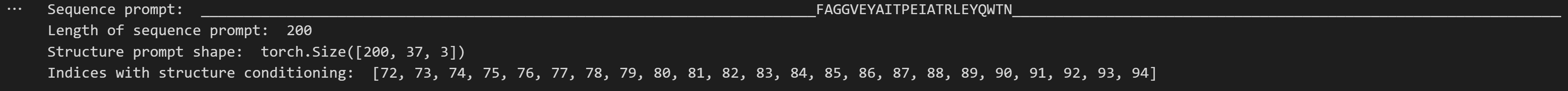
Description automatically generated with medium confidence



The shape of the array is (n\_residues, 37, 3) where n\_residues is the number of residues in the protein and 37 is the number of possible distinct atoms that may be present across all amino acids (e.g. the first three atoms are the N, C-alpha, and C atoms corresponding to the protein backbone). The 3 corresponds to the x, y, and z coordinates of each atom.

### 3. Masking the protein

Next we will create a prompt for the model to use that’s 200 characters with the motif we chose will be the conditioning and the rest of the characters are masked.

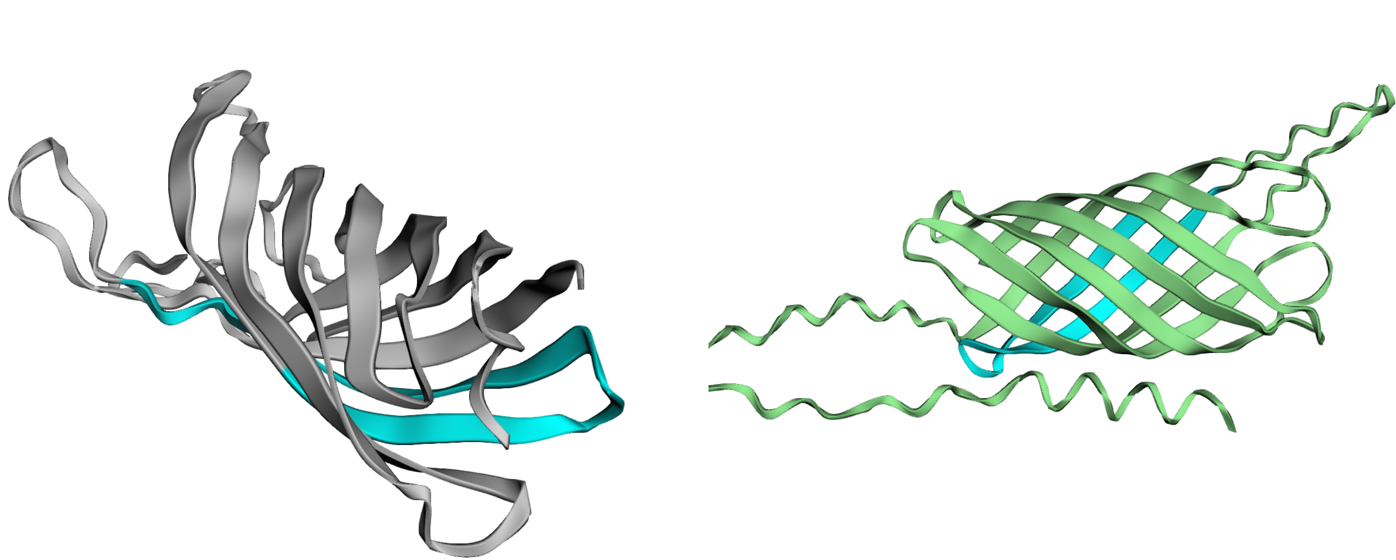


What we can see in the code is the sequence is masked until index 72 where we insert the motif sequence and mask the rest of the sequence completing 200 chars. And do the same for the structure. Finally, we insert it into the ESMProtein class that ESM3 can use.

Now we can generate the sequence and the structure like so:

A screenshot of a computer

Description automatically generated



Where the left structure is the original ( gray is masked and blue is the motif )

And the right structure is the generated structure (green is generated blue is the motif)

## Testing on csv data

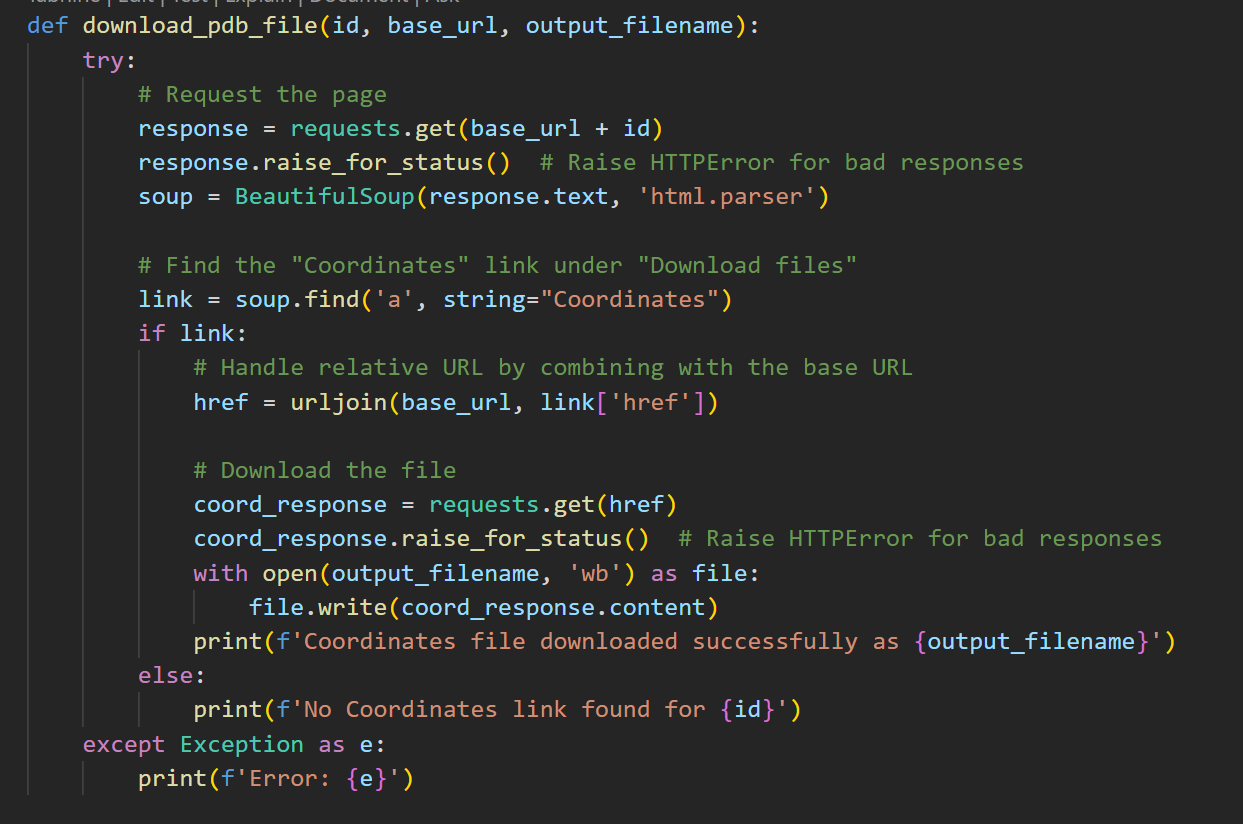
Now that our pipeline is working we can try and predict the RMSD value for all strands in the ‘OMBB\_data.csv’ file provided to us.

### 1. Getting Data

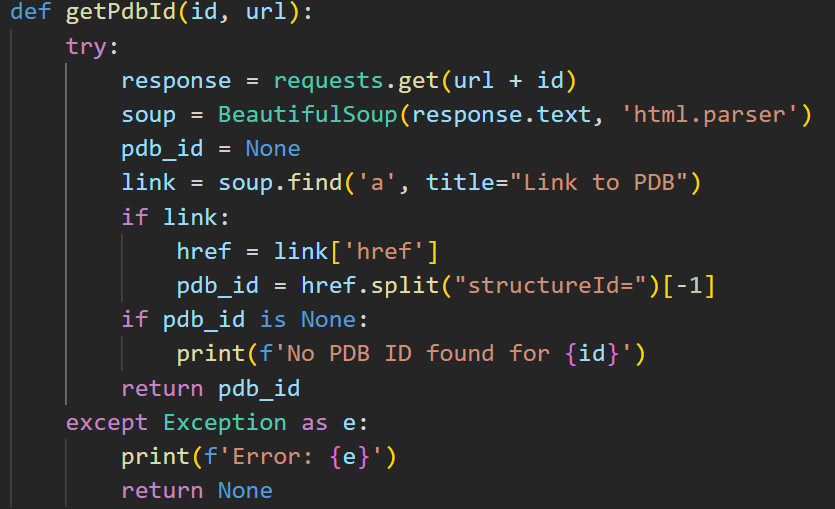
We wrote a script to get all pdb files from <http://prodata.swmed.edu/ecod/>

We use a web crawler to find the link to download the file and we save it locally.

Using this on all the id’s we get a folder of all pdb files.

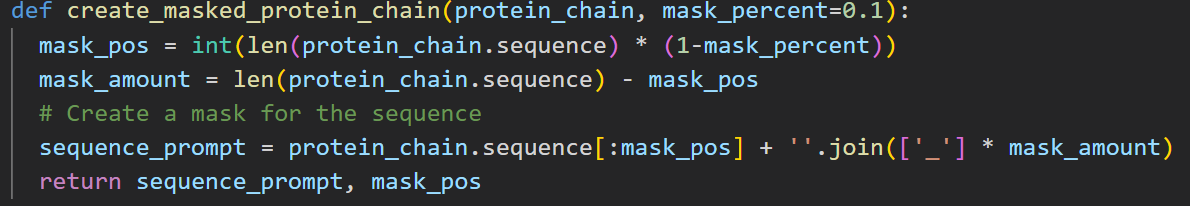


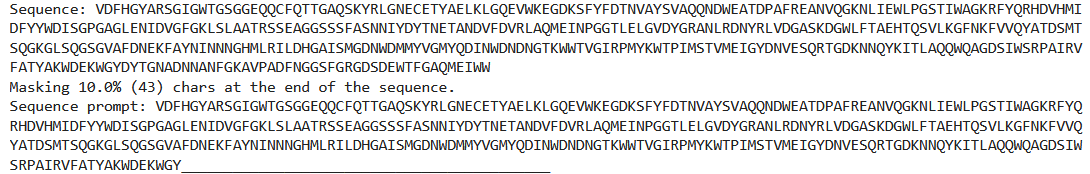
Sometimes there are errors in the pdb files. In those cases we get the matching id from the pdb ( <https://www.rcsb.org/> ) and load the pdb file from there.



### 2. Mask a percentage of the protein

Now that we have the pdb file and we have loaded the protein into the ESM class ‘ProteinChain’ we want to mask the end of the sequence.





### 2. Generate the structure

Now we use the following function to get out prediction

A screen shot of a computer code

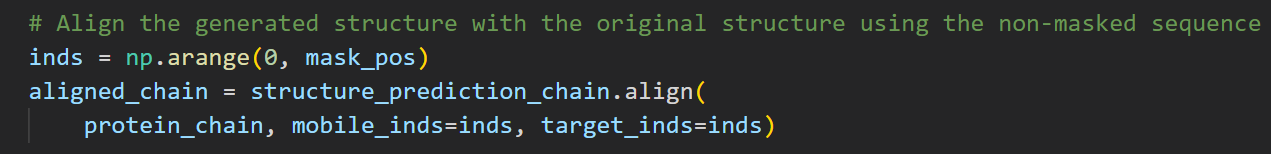
Description automatically generated

Here we complete the sequence and use it to generate the structure

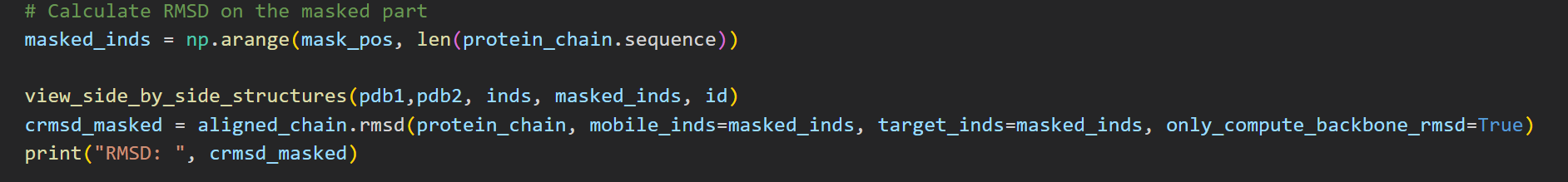
### 3. Calculate RMSD

Now we wish to align the ground truth structure with the generated structure

For this we simply use a function from ESM that aligns the structure based on the indices before the mask.



Now that the structures are aligned we can use another library function to calculate the RMSD of the generated part.



Where the RMSD is calculated as:

Where is the distance between atom i and the ground truth structure calculated only on the backbone atoms.

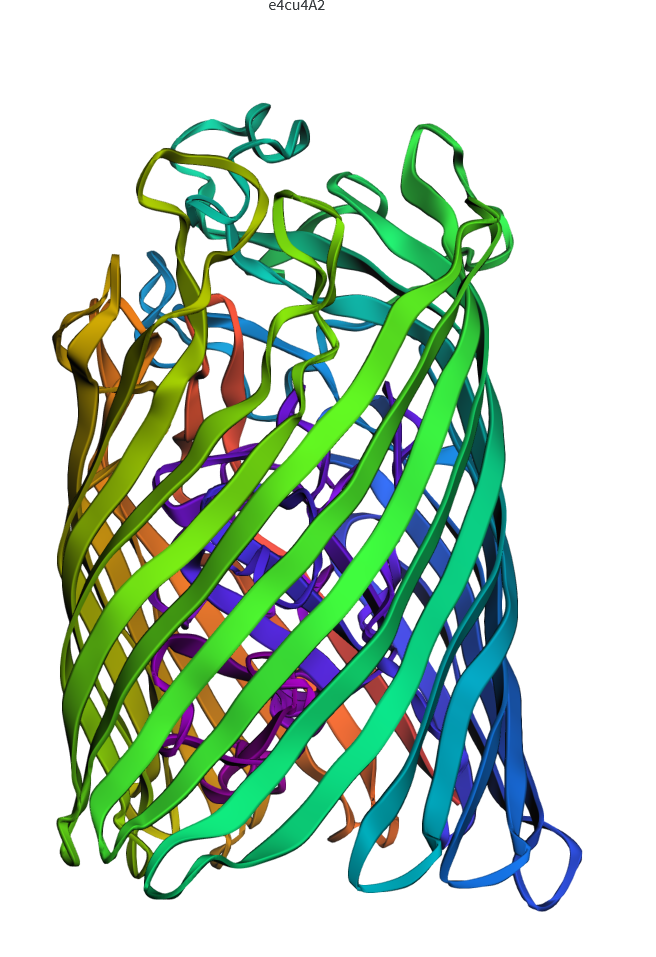
### 4. Views of the protein

We saved multiple views for each protein

You can view each protein in wandb : <https://api.wandb.ai/links/radiostars/ya2378wy>

For example, for protein id: e4cu4A2

The grey area is the part of the sequence we masked.

A blue and white spiral object

Description automatically generated

This is the generated protein superimposed on the ground truth structure.

A close-up of a blue and green ribbon

Description automatically generated

This is the original protein side by side with the generated protein. The green is the generated motif.

A blue and green spirals

Description automatically generated

### 5. Results:

We created a new csv with all the RMSD values for each protein.

You can view all the results in wandb : <https://api.wandb.ai/links/radiostars/ya2378wy>

And we have calculated the min, max, std and mean of the RMSD values.

Min = 0.2628

Max = 11.69

Mean = 2.432

Std = 2.178

We also created a histogram of the RMSD values:

A graph with blue squares

Description automatically generated with medium confidence

And a scatter plot for each id. Points colored in yellow have highest RMSD while points in black have small RMSD

A graph of different colored dots

Description automatically generated