**Course: Advanced Bioinformatics**

**Module title: Protein Structure**

**Module no. : 197**

The protein structure modules provide tools for representing and manipulating 3D biomolecular structures. It particularly focuses on protein structure comparison.

The following algorithms have been implemented and included in BioJava.

FATCAT algorithm for flexible and rigid body alignment is used in BioJava. The standard Combinatorial Extension (CE) algorithm is also employed. A new version of CE that can detect circular permutations in proteins is part of the API. These algorithms are used to provide the RCSB Protein Data Bank (PDB), Protein Comparison Tool as well as systematic comparisons of all proteins in the PDB on a weekly basis.

Parsers for PDB and mmCIF file formats allow the loading of structure data into a reusable data model. This feature is used by the SIFTS project to map between UniProt sequences and PDB structures. Information from the RCSB PDB can be dynamically fetched without the need to manually download data. For visualization, an interface to the 3D viewer Jmol http://www.jmol.org/ is provided. The team claims that work is underway to improve interaction with the RCSB PDB viewers.

Below is an outline of the code to initialize a window that will display and compare two protein sequences. Please bear in mind that this is just an outline of the code. To make this work one will need to import the correct found in the "org.biojava.bio.structure" package and add also handle exceptions by using a try-catch block.

String name1 = "4hhb.A";

String name2 = "4hhb.B";

AtomCache cache = new AtomCache();

Structure structure1 = null;

Structure structure2 = null;

StructureAlignment algorithm =

StructureAlignmentFactory.getAlgorithm(FatCatRigid.algorithmName);

structure1 = cache.getStructure(name1);

structure2 = cache.getStructure(name2);

Atom[] ca1 = StructureTools.getAtomCAArray(structure1);

Atom[] ca2 = StructureTools.getAtomCAArray(structure2);

FatCatParameters params = new FatCatParameters();

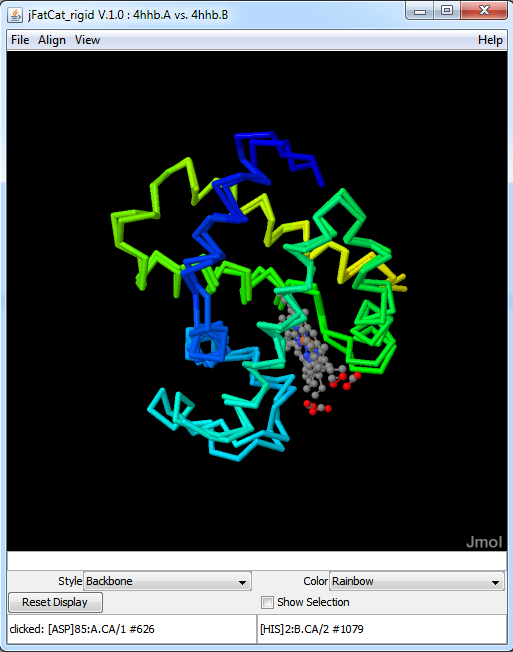
AFPChain afpChain = algorithm.align(ca1,ca2,params);

afpChain.setName1(name1);

afpChain.setName2(name2);

StructureAlignmentDisplay.display(afpChain, ca1, ca2);

The code aligns the two protein sequences "4hhb.A" and "4hhb.B" based on the FATCAT rigid algorithm

[](https://en.wikipedia.org/wiki/File:This_window_shows_two_proteins_with_IDs_%224hhb.A%22_and_%224hhb.B%22_aligned_against_each_other.png)

This window shows two proteins with IDs "4hhb.A" and "4hhb.B" aligned against each other. The code is given on the left side. This is produced using BioJava libraries which in turn uses Jmol viewer.[[4]](https://en.wikipedia.org/wiki/BioJava#cite_note-Jmol-4) The FATCAT[[17]](https://en.wikipedia.org/wiki/BioJava#cite_note-fatcat-17) rigid algorithm is used here to do the alignment.