POZNAN UNIVERSITY OF TECHNOLOGY

EUROPEAN CREDIT TRANSFER AND ACCUMULATION SYSTEM (ECTS)

COURSE DESCRIPTION CARD - SYLLABUS

Course name
Structural Bioinformatics [S1Bioinf1>BSTR]

Course

Field of study Bioinformatics	Year/Semester 3/5
Area of study (specialization) –	Profile of study general academic
Level of study first-cycle	Course offered in Polish
Form of study full-time	Requirements compulsory

Number of hours

Lecture 15	Laboratory classes 30	Other 0
Tutorials 0	Projects/seminars 0	

Number of credit points

4,00

Coordinators

dr hab. inż. Maciej Antczak prof. PP
maciej.antczak@put.poznan.pl

Lecturers

Prerequisites

A student starting this module should have basic knowledge of molecular biology and skills in the use of web servers and databases available on the Internet, programming in Python, designing relatively simple algorithms, and versioning source codes using the Git tool (e.g., on the GitHub platform). Moreover, the student should show attitudes such as honesty, persistence, creativity, and respect for other people. Finally, he/she should be able to obtain information from the indicated sources, often in English.

Course objective

The course aims to introduce students to the basic concepts and widely used algorithms, computational methods, and databases used during the study, analysis, and interpretation of the structures of biological molecules. In particular, this includes: 1. Provide students the principal knowledge from the field of structural bioinformatics with particular emphasis on computational methods and algorithms for prediction, analysis, visualization, and comparison of molecular structures. 2. Providing students the basic knowledge about specialized databases, available on the Internet, storing information about the structures of biological molecules, and developing the skills of their practical use. 3. Developing students" ability to define the most appropriate paths of conduct, during which the available structural data are analyzed and interpreted, to find an answer within the research problem under consideration. 4. Developing students" skills in the development of programs that successfully use application programming interfaces (APIs) of widely used databases and web servers, and protection of them against unavailability of external resources. 5. Developing students" skills in the field of homologous modeling, molecular dynamics, and docking of protein structures.

Course-related learning outcomes

Knowledge:
As a result of the conducted course, the student:
1. Knows and understands the basic properties of biological molecules and biological processes that occur with their participation, and bases their interpretation on empirical foundations, using mathematical methods, including statistical methods, and machine learning.
2. Knows and understands concepts in the field of physics useful for the formulation and solving of simple bioinformatics problems, including the basic properties of molecular structures and their hierarchical organization, which are crucial in the analysis of intermolecular interactions.
3. Knows and is able to apply basic methods, algorithms, and tools to design and develop software dedicated to the integration and interpretation of molecular data in the process of solving bioinformatics tasks.
4. Knows and understands the basic concepts in the field of structural bioinformatics, including molecular modeling and docking.

Skills:
As a result of the conducted course, the student:
1. Can obtain information from scientific literature, specialized databases storing information about molecular structures, and other properly selected sources, e.g., open-source software repositories, published on the GitHub platform, providing algorithms and tools that solving problems in structural bioinformatics.
2. Can integrate various kinds of biological data and interpret the information obtained on their basis, as well as accurately conclude and formulate and justify his/her opinions based on the results of computational experiments.
3. Can design and produce specialized data flows integrating widely used databases and web servers to solve problems inspired by biological applications and evaluate their computational efficiency and practical usability.
4. Can easily adapt basic IT techniques and tools when designing algorithms and developing software dedicated to prediction and analysis of biological molecule structures.

Social competences:
As a result of the conducted course, the student:
1. Recognizes the need for continuous training and improvement of own competences in the field of structural bioinformatics.
2. Can function and cooperate in a group, playing various roles in it, and is able to properly define priorities for the implementation of a task set by himself or others.
3. The practical nature of the course allows the student to develop entrepreneurial skills.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
The knowledge and skills acquired during the course (considering both the lecture and the laboratories) are verified by a 60-minute exam test. The test is carried out during the examination session and the

students solve it on their own. The exam includes about 20 questions (open and closed, differently marked). Passing threshold: 50% of points.
The skills acquired during the laboratories are verified based on partial grades obtained from the mini-projects carried out during the classes. To pass the laboratories it is necessary to realize and present every considered mini-project. The final grade is the average of all partial grades.

Programme content

The lecture program covers the following topics:
* basic information about the course being carried out and the methods of its completion,
* basic terms and concepts associated with structural bioinformatics of biological molecules,
* central dogma of molecular biology (DNA, RNA, protein),
* structural and functional analysis of biological molecules,
* hierarchical structural organization of proteins and nucleic acids,
* primary structure: chemical properties of amino acids, FASTA format, local/global sequence alignment, and corresponding algorithms, substitution matrices, PSI-BLAST,
* polypeptide chain geometry, Phi/Psi angle, Ramachandran plot, covalent and non-covalent chemical bonds in proteins,
* secondary structure: hydrogen bonding, extraction of the secondary structure from the 3D structure (e.g., DSSP), methods of predicting the secondary structure, contact maps,
* tertiary structure: experimental determination methods of tertiary structures of biological molecules, formats: PDBx/mmCIF and PDB, the definition of the protein folding problem from the sequence, the relationship between the 3D fold and the function of the molecule in the cell, automated prediction methods of 3D structures of biological molecules, homologous modeling of proteins (e.g., MODELLER), molecular dynamics (e.g., GROMACS),
* classification of tertiary structures of proteins (e.g., SCOP, CATH),
* quaternary structure,
* approaches for the structural alignment of proteins and RNAs,
* quality assessment of 3D structures of biological molecules, widely used assessment measures, e.g., RMSD, GDT_TS,
* the most popular tools for the visualization of 3D structures of biological molecules, e.g., LiteMol, JSMol, Mol * (different ways of the structure presentation, e.g., Cartoon, Ball and Stick),
* widely used repositories storing 3D structures of biological molecules and associated additional data (e.g., Protein Data Bank, ASTRAL, UniProt),
* review of the most popular web servers solving structural bioinformatics challenges (e.g., I-TASSER),
* protein complexes and widely used molecular docking approaches (e.g., HADDOCK),
* libraries supporting the development of software dedicated to the analysis and processing of biological molecule structures (e.g., biopython).
Laboratories conducted in the form of 15 two-hour classes are taking place in the computer lab. During the laboratory classes, students individually or in groups of two carry out 6-7 mini-projects related to structural bioinformatics. The projects are practical and include the implementation of relatively simple scripts for solving the indicated problems and conducting the necessary experiments. Students have 2-3 weeks to implement each project, depending on its complexity. Progress is continuously tracked by the tutor during the classes. The list of proposed projects is made available before the beginning of the semester and periodically updated according to the current state of development of structural bioinformatics.

Course topics

none

Teaching methods

1. Lecture: slide show presentations supplemented, if necessary, with additional examples presented on the blackboard.
2. Laboratory classes: practical exercises at the computer carried out according to a specific scenario, implementation of relatively simple programs and network services, designing and conducting computational experiments and analysis of the obtained results, discussion of the applied solutions, and presentation of the obtained results.

Bibliography

Basic
1. A. M. Lesk, "Introduction to bioinformatics", 2002, Oxford University Press.
2. J. Xiong, "Essential bioinformatics", 2006, Cambridge University Press; 1st edition.
3. A. D. Baxevanis i B. F. F. Ouellette, "Bioinformatics: A Practical Guide to the Analysis of Genes and Proteins", 2004, Wiley-Interscience; 3rd edition.
Additional
1. J. Gu, P. E. Bourne, "Structural Bioinformatics", 2019, Wiley-Blackwell; 2nd edition.
2. I. Eidhammer,W. R. Taylor,J. Inge, "Protein Bioinformatics", 2009, Wiley India Pvt Ltd; 1st edition.
3. M. Gromiha, "Protein Bioinformatics: From Sequence to Function", 2010, Academic Press; 1st edition.
4. C. Setubal, J. Meidanis, "Introduction to Computational Biology", 1997, PWS Publishing; 1st edition.
4. A. M. Lesk, "Introduction to Protein Architecture: The Structural Biology of Proteins", 2001, Oxford University Press; 1st edition.
5. S.L. Salzberg, D.B. Searls, S. Kasif, "Computational Methods in Molecular Biology", 1998, Elsevier Science.

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	45	2,00
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	55	2,00