COURSE DESCRIPTION CARD - SYLLABUS

Course name
Biological Database Applications [S2Bioinf2>BABD]

Course

Field of study
Bioinformatics

Year/Semester
1/1

Area of study (specialization)
–

Profile of study
general academic

Level of study
second-cycle

Course offered in
Polish

Form of study
full-time

Requirements
compulsory

Number of hours

Lecture
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

5,00

Coordinators

dr inż. Anna Leśniewska
anna.lesniewska@put.poznan.pl

Lecturers

dr inż. Anna Leśniewska
anna.lesniewska@put.poznan.pl

Prerequisites

The student starting this course should have basic knowledge regarding basics of programming and operating systems. The student should have the ability to build queries in SQL language in order to be able to communicate with the database. The student should understand the need to expand their competences and be ready to work in a team. The student should have skills that are necessary to acquire information from given sources of information. Moreover, the student should show such attitudes as honesty, responsibility, perseverance, curiosity, creativity, personal culture, and respect for other people.

Course objective

Provide students with basic knowledge of database technology necessary for the correct design, use and implementation of database systems and application. Develope students' skills in solving problems arising in the management of database systems.

Course-related learning outcomes

Knowledge:
- The student has knowledge about the life cycle of database systems
- The student knows the tools and techniques in the field of data processing and analysis in databases

used to solve bioinformatics tasks

Skills:
- is able to use materials in the form of literature, teaching materials and technical documentation of database systems in Polish and English
- is able to apply the known tools and techniques of advanced data analysis in the database to solve biological problems
- is able to perform research tasks under the guidance of the teacher, planning and using analytical methods on the data in the database
- is able to formule and tests hypotheses related to bioinformatics problems and verifies them during the analysis
- is able to use the acquired skills to create software, e.g. in the form of generating forms and reports, as well as software for managing and manipulating data in a database
- is able to programme in SQL and PL/SQL language

Social competences:
- understands that knowledge in the field of databases is changing very dynamically and systematically update your knowledge and skills is needed
- is ready to perform professional roles responsibly, taking into account the upholding of the ethos of the profession, and to adhere to the principles of professional ethics and act to uphold these principles

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

The knowledge acquired during the lecture is verified by the test carried out during the 15 lecture, consisting of 10-15 questions (test and open), with diversified scores. Passing threshold: 50% of the total score. Topics, based on which the questions are developed, will be sent to students by university e-mail. The skills acquired during the laboratory classes are verified on the basis of a list of grades given during the semester in the form of a weighted average (1/2 implementation of exercises in class) and preparation and presentation of a database application by the student (1/2 of the project grade).

Programme content

Lectures cover the following topics:
- introduction to the database system and specificity of biological databases
- conceptual modeling
- relational model of data
- programming in PL/SQL language and using it in bioinformatics analysis
- user authentication methods
- principles of granting permissions and methods of authorization in the database
- index structures in databases
- software interfaces
- research problems in working with biological databases
- visualization of biological data stored in a database
- NoSQL databases and biological databases

Laboratory exercises are a practical use of the content presented in the lecture and cover the following issues:
- SQL language for advanced bioinformatics analyzes (GROUP BY and HAVING grouping, ROLLUP and CUBE cube and semi-cube, transformation of row data to PIVOT and UNPIVOT column layout)
- programming in PL / SQL - selected elements (language concept, stored procedures, functions and packages)
- user authentication methods
- principles of granting permissions and methods of authorizing database operations
- Indexes
- selected software interfaces to relational databases (JDBC, R, Python) and using it in working with biological databases
- designing a database application in the Oracle APEX environment
- working with a document database on the example of MongoDB

Course topics

The classes present key topics related to working with and analyzing biological data stored in relational and non-relational databases:
Advanced SQL
- Grouping (GROUP BY) and filtration (HAVING): key statements for data aggregation. GROUP BY groups rows with the same values, and HAVING filters the results after aggregation.
- Cube and semi-cube (ROLLUP and CUBE): Allows you to create complex multidimensional reports by generating summaries at different levels of the hierarchy.
- Data transformation (PIVOT and UNPIVOT): PIVOT converts data from rows to columns, which makes analysis easier. UNPIVOT works the other way around, transforming columns to rows.
PL/SQL
- Extension of SQL with Stored Procedures: Creating procedures, functions and packages allows you to fragment the code and reuse it.
Authentication and Authorization
- Authentication methods and granting permissions: Important for data security.
Database access technologies using different programming environments:
- JDBC: Allows you to connect Java to databases.
- R and Python: Widely used in bioinformatics for data analysis.
Designing applications in Oracle APEX
- Oracle Application Express: A tool for creating simple web applications based on cooperation with the database.
Working with NoSQL databases
- Document databases on the example of MongoDB as a representative of the NoSQL trend databases, which stores data in the JSON document format. It enables flexible data modeling and effective management of large data sets.

Teaching methods

1. Lecture: multimedia presentation augmented with additional examples given on the whiteboard.
2. Laboratory exercises: solving tasks, practical exercises, teamwork, multimedia presentation.

Bibliography

Basic:
1. Garcia-Molina H., Ullman J.D., Widom J., Implementacja systemów baz danych, WNT, 2003
2. Ullman J.D. , Widom J., Podstawowy wykład z systemów baz danych, WNT, W-wa, 2000
3. Elmasri R., Navathe S., Wprowadzenie do systemów baz danych, Helion, (4th Edition), 2005
4. Wrembel R., Wieczerzycki W., Projektowanie aplikacji baz danych Oracle. NAKOM.
5. Feuerstein S., Pribyl B., Dawes C., Język Oracle PL/SQL. Leksykon podstawowy, Helion
6. Matloff N., The Art of R Programming, No Starch Press, 2011

Additional:
1. Braun, W., & Murdoch, D. (2007). A First Course in Statistical Programming with R. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511803642
2. Quick John M., Analiza statystyczna w środowisku R dla początkujących, Helion

Breakdown of average student's workload

	Hours	ECTS
Total workload	100	4,00
Classes requiring direct contact with the teacher	60	2,50
Student's own work (literature studies, preparation for laboratory classes/ tutorials, preparation for tests/exam, project preparation)	40	1,50