COURSE DESCRIPTION CARD - SYLLABUS

Course name
Laboratory of Bioinformatics Algorithms [S1Bioinf1>PAB]

Course

Field of study
Bioinformatics

Year/Semester
2/4

Area of study (specialization)
–

Profile of study
general academic

Level of study
first-cycle

Course offered in
Polish

Form of study
full-time

Requirements
elective

Number of hours

Lecture
30

Laboratory classes
30

Other
0

Tutorials
0

Projects/seminars
0

Number of credit points

4,00

Coordinators

dr inż. Marcin Radom
marcin.radom@put.poznan.pl

Lecturers

Prerequisites

Students starting this module should have basic knowledge of algorithms, data structures and computational complexity classes. Additionally, they should be able to construct algorithms, implement them and examine the quality of the results returned by them. It is required that the student has basic object-oriented programming skills in C++ and/or Java. Moreover, in terms of social competence, a student should demonstrate such attitudes as conscientiousness, responsibility, regularity, creativity, accountability, perseverance and cognitive curiosity.

Course objective

1. Broadening the knowledge of algorithms and applying them to solve bioinformatics problems. 2. Develop in students the ability to analyze the complexity of the problems and construct effective solutions. 3. To get students acquainted with scientific publications that provide the current state of knowledge in researched bioinformatics problems. 4. To provide students with knowledge of C# language. 5. To create a common application with the possibility to compare and visualize the results of the designed algorithms. 6. To develop students" object-oriented programming skills for algorithm implementation.

Course-related learning outcomes

Knowledge:
As a result of the course, the student knows:
1. issues in bioinformatics algorithms, basic and advanced elements of C# language,
2. principles of object-oriented programming in C# language which he/shee can apply to bioinformatics problems.

Skills:
As a result of the course, the student will be able to:
1. design and develop software for bioinformatics applications according to a given specification, using appropriate methods, techniques and tools.

Social competences:
Passing the course means that the student:
1. understands the need for lifelong learning and improving competence due to new discoveries and methods in bioinformatics sciences and the continuous development of programming languages.

Methods for verifying learning outcomes and assessment criteria

Learning outcomes presented above are verified as follows:

Learning outcomes presented above are verified as follows:
Formative assessment:
a) in the scope of lectures, verification of the assumed educational effects is realized by answers to questions concerning the material discussed during previous lectures
b) in the scope of laboratory, verification of the assumed educational effects is realized by the evaluation of the development of applications needed to run and visualize the designed algorithms, evaluation of knowledge and skills related to the implementation of the laboratory tasks through the projects completed with the report and presentation and comparison of the results generated by the proposed algorithms.
Summative assessment:
a) in the scope of lectures, verification of the assumed educational effects is realized by evaluation of knowledge and skills demonstrated on a written colloquium in the form of multiple choice test and open questions. The test consists of 10-15 questions with a total value of 20-30 points distributed depending on the difficulty of the question. Students receive a passing grade when a minimum of half of the points is scored.
b) in the scope of laboratory, the verification of the assumed learning outcomes is realized by the independent design and implementation of solutions to the known bioinformatics problems, based on the applications developed during the classes. Each solution must be described in reports together with conclusions including further possibilities of improvement of implemented algorithms. Laboratory assignments consist of solving the problems studied. For each student receives a grade depending on the quality of the presented solution and the time to complete the project. The students receive a positive grade from the laboratory if they hand in all projects and the average of these grades is not less than 3.0. The final grade is also affected by the colloquia conducted during the laboratory.
The activity during the classes is rewarded with additional points, in particular for the discussion of additional aspects of the problem, the effectiveness of the application of the knowledge gained when solving the problem and for the quality of the proposed solutions and their clear presentation to the rest of the group.

Programme content

The lecture program covers topics in algorithm theory, bioinformatics problems, and the C# language. Laboratory exercises are conducted as fifteen two-hour classes held in the laboratory. The first class is designed to familiarize students with the use of the laboratory and to pass the exercises.

Course topics

The lecture program covers topics in algorithm theory, bioinformatics problems, and the C# language. During the lectures students get acquainted with:
- syntax, objects, programming styles in C# language, creating window programs e.g. in Visual Studio environment,
- classes of computational complexity, examples of exact and heuristic algorithms for bioinformatics

problems,
- various bioinformatics problems.
Laboratory exercises are conducted as fifteen two-hour classes held in the laboratory. The first class is designed to familiarize students with the use of the laboratory and to pass the exercises. The program of laboratory classes includes the following issues:
- implementation of examples presented in lectures,
- writing of independent programs as part of proposals of solutions to the investigated problems,
- implementation of larger applications allowing comparison and visualization of the proposed solutions,
- exercises on proper programming practices needed for effective implementation of the proposed solutions. Laboratory projects include bioinformatics issues.

Teaching methods

1. Lecture: multimedia presentation and additional examples given on the blackboard as needed.
2. Laboratory exercises: solving tasks, implementing solutions, practical exercises, iterative development of created software, teamwork.

Bibliography

Basic
1. Wprowadzenie do algorytmów, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Wydawnic-twa Naukowo - Techniczne, 2004.
Additional
1. Algorytmy i struktury danych, L. Banachowski, K. Diks, W. Rytter, Wydawnictwa Naukowo - Techniczne, 2006.
2. Złożoność obliczeniowa algorytmów i problemów szeregowania zadań, Jacek Błażewicz, Wydawnictwo Politechniki Po-znańskiej, 1979.
3. Złożoność obliczeniowa problemów kombinatorycznych, Jacek Błażewicz, Wydawnictwa Naukowo-Techniczne, 1988.

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