

Degree programme	Type	Course
Bioinformatics	OB	1

Contact lecturer

Name : Elisa Ruth Heymann Pignolo

Email : elisa.heyman@uab.cat

Teaching staff

Lidia Garrido Sanz

Group languages

You can consult this information at the [end](#) of the document.

Prerequisites

For the general development of the course, it is recommended to have a B2 level, or equivalent, of the English language.

For this module, it is recommended to have basic notions of computer usage in Linux (i.e., knowledge of common user tools and file manipulation).

Objectives

General objectives of this module are the application of the core tools and basic techniques for development in this area of knowledge. Provide skills to successfully assume the adaptation to changing technologies and new paradigms emerging in this interdisciplinary field.

Learning outcomes

- CA01 (Programming efficiently and accurately to solve bioinformatic problems of varying complexity.) Programming efficiently and accurately to solve bioinformatic problems of varying complexity.
- CA02 (Propose bioinformatics solutions to complex problems by participating in multidisciplinary teams.) Propose bioinformatics solutions to complex problems by participating in multidisciplinary teams.
- KA01 (Describe the fundamental concepts of programming, algorithmics, and information organisation that are applied in bioinformatics.) Describe the fundamental concepts of programming, algorithmics, and information organisation that are applied in bioinformatics.
- KA02 (Identify the main algorithmic schemes and their application cases in bioinformatics.) Identify the main algorithmic schemes and their application cases in bioinformatics.
- KA03 (Explain the fundamentals and procedures of the methods of access, analysis and data mining in

bioinformatics, highlighting their usefulness for the efficient management and interpretation of large volumes of biological sequences.) Explain the fundamentals and procedures of the methods of access, analysis and data mining in bioinformatics, highlighting their usefulness for the efficient management and interpretation of large volumes of biological sequences.

- SA01 (Solve problems of a biological nature by applying programming concepts and the use of specialised bioinformatics libraries (such as BioPython or Bioconductor), guaranteeing the reliability of the code and the biological interpretation of the results.) Solve problems of a biological nature by applying programming concepts and the use of specialised bioinformatics libraries (such as BioPython or Bioconductor), guaranteeing the reliability of the code and the biological interpretation of the results.
- SA02 (Apply the main algorithmic schemes and some of their variants in the resolution of bioinformatic problems.) Apply the main algorithmic schemes and some of their variants in the resolution of bioinformatic problems.
- SA03 (Integrate access to large biological databases with local infrastructures and cloud systems, combining the necessary algorithms.) Integrate access to large biological databases with local infrastructures and cloud systems, combining the necessary algorithms.

Contents

1. Linux (commands and shell scripting)

Basic commands, user management, software management, and file system

Text processing tools and data manipulation

Redirections, pipes, and filters

Shell scripting in Bash

2. Programming languages

Introduction to R

Introduction to programming in Python in Bioinformatics

Variables, expressions, data types, operators, programming constructs, and contexts

Code reutilization: functions, modules, and subroutines

Recursive programming

Input/Output

Code debugging

3. Data structures and data processing

Basic data structures (including strings, lists, tuple, sets, and dictionaries)

Nested data structures and objects

Trees and graphs

Modelling and representing bioinformatics data

Basic bioinformatics data formats (including FASTQ, SAM, VCF)

Regular expressions

4. Bioinformatics libraries and tools

Data visualisation tools

Introduction to Biopython

Introduction to NumPy and Pandas

Learning activities and methodology

Title	Hours	ECTS	Learning outcomes
Classroom work	20	0.8	
Problem solving (in class)	14	0.56	
Regular general work on the deliverables definition and materials given	83	3.32	
Work in the computing lab	12	0.48	
Performing lab work from recommending reading	15	0.6	

The methodology combines classroom work, supervised in-class problem solving, unsupervised work in the computing lab, homework from recommended readings and independent study student. For the delivery of the students work the virtual platform will be used.

Restricted Use of AI: For this course, the use of Artificial Intelligence (AI) technologies is permitted in a limited manner. For lab exercises and deliveries, students may use AI tools for general questions, but they are NOT allowed to submit code generated by these tools. In the lab exercises reports, students must clearly identify the assistance received from these tools, specify which tools were used, and show the prompts used. A lack of transparency in the use of AI will be considered a breach of academic integrity and may result in partial or total penalties on the activity grade, or more serious sanctions.

Annotation: within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Assessment

Continuous assessment activities

Title	Weight	Hours	ECTS	Learning outcomes
Laboratory work	30%	3	0.12	CA01, CA02, KA01, KA02, SA01, SA02, SA03
Evaluation of work done during the module, presented	10%	1	0.04	CA01, CA02, KA01, KA02

by the user				
Final Exam	60%	2	0.08	CA01, CA02, KA01, KA02, KA03, SA02

The methodology will combine classroom work, problem solving in the classroom, unsupervised work done in the computing lab and individual work from recommended readings. It will make use of the virtual platform and will make references to selected publications related to the thematic blocks. None of the individual assessment activities will account for more than 50% of the final mark.

Retake exam

To be eligible for the retake process, the student should have been previously evaluated in a set of activities equaling at least two thirds of the final score of the module. The teacher will inform the procedure and deadlines for the retake process. Please note that activities performed in class cannot be made up.

Not valuable

The student will be graded as "Not Valuable" if the weight of the evaluation is less than 67% of the final score.

This subject/module does not offer a single assessment evaluation.

Any irregularity committed during an assessment activity (academic misconduct, plagiarism, or improper use of AI, unless such use is expressly authorized in the course syllabus) that that could significantly alter the grade will result in that activity being graded as 0. If the course syllabus stipulates that obtaining a minimum mark in this assessment is an essential requirement to pass the course, or if multiple irregularities occur in the assessment activities of the same course, the final grade for the course will be 0. Furthermore, disciplinary proceedings may be initiated against any student who incurs any of these irregularities.

Bibliography

- Bell, Charles; Kindahl, Mats; Thalmann, Lars. "MySQL High Availability". O'Reilly, 2010.
- Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J., Rapp, B. A. & Wheeler, D. L. (2002).
- GenBank. *Nucl. Acids Res.*, 30(1):17-20. URL <http://nar.oupjournals.org/cgi/content/abstract/30/1/17>.
- Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. & Bourne, P. E. (2000). The Protein Data Bank. *Nucleic Acids Res.*, 28(1):235-242.
- Bessant, C., Shadford, I., Oakley, D. "Building Bioinformatics Solutions with Perl, R and MySQL", Oxford University Press, 2009
- Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M.-C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S. & Schneider, M. .The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res.*, 31(1):365-370.
- Christiansen, P., Wall, L., Orwant, J., "Programming Perl". 4th Edition, O'Reilly, 2012
- Mäkinen et al., *Genome-Scale Algorithm Design: Biological Sequence Analysis in the Era of High-Throughput Sequencing*. Cambridge Univ. Press, 2015.
- Matloff, N., "The Art of R Programming". No Starch Press Inc., 2011
- Lutz, M., "Learning Python", O'Reilly, 5th edition, 2013
- Siever, E., Figgins, S., "Linux in a nutshell" O'Reilly 2009.
- Sobell, M., "A Practical Guide to Linux. Commands, editors and shell programming". Prentice Hall, 2009.
- Tindall, James., *Begining Perl for Bioinformatics*. O'Reilly 2012.

Recommended websites

- <http://mscbioinformatics.uab.cat>
- <https://cv.uab.cat>

Search for bioinformatics and computer science topics in UAB library e-book resources:

- <http://www.uab.cat/biblioteques/trobador>
- <http://pagines.uab.cat/bctdigital>

Software

Linux (Ubuntu, Bash, linux-tools, etc)

Python 3.x

Jupyter Notebook / PyCharm

Matplotlib/Seaborn

Numpy/Pandas

Course groups and languages

The information provided is provisional until November 30. After this date, you will be able to consult the language of each group through this [link](#). To access the information, you will need to enter the course CODE

Type of teaching	Group	Language	Semester	Shift
(TEm) Theory (master)	1	English	first semester	morning-mixed
(PLABm) Practical laboratories (master)	1	English	first semester	morning-mixed