UNIVERSIDAD AUTÓNOMA DE MADRID

18245 - PROGRAMMING TOOLS IN BIOCHEMISTRY AND MOLECULAR BIOLOGY

This is a non-sworn translation intended to provide students with information about the course

Information of the subject

Code - Course title: 18245 - PROGRAMMING TOOLS IN BIOCHEMISTRY AND MOLECULAR BIOLOGY

Degree: 531 - Graduado/a en Bioquímica

Faculty: 104 - Facultad de Ciencias

Academic year: 2023/24

1. Course details

1.1. Content area

-

1.2. Course nature

Optional

1.3. Course level

Grado (EQF/MECU 6)

1.4. Year of study

4

1.5. Semester

First semester

1.6. ECTS Credit allotment

6.0

1.7. Language of instruction

Español

1.8. Prerequisites

-

1.9. Recommendations

-

1.10. Minimum attendance requirement

-

1.11. Subject coordinator

Luis Peso Ovalle

https://autoservicio.uam.es/paginas-blancas/

1.12. Competences and learning outcomes

1.12.1. Competences

-

1.12.2. Learning outcomes

-

1.12.3. Course objectives

-

1.13. Course contents

Topic 0. Presentation of the course. Objectives, structure and teaching and evaluation method of the course. Contents. Introduction to the command line and text editors for programming.

BLOCK I. Python
Topic P1. Variables, expressions, operators in python. Types. Variables. Operators and precedence. Expressions. Data structures.

Topic P2. Conditional execution, recursion, iteration. Logical operators. Execution flow (if). Conditional and alternative execution. Recursion. Iteration (for, while).

Topic P3. Reading and writing data. Directory tree structure. Reading files from disk. Parsing of files. File writing. Handling of arguments passed by command line.

Topic P4. Functions and modularity. Functions: arguments, return values, void functions. Composition of functions. Execution flow. Local and global variables. Import modules.

Topic P5. Regular expressions. What are regular expressions and what are they for? Regular expressions in Python.

Topic P6. Object-oriented programming. Classes, functions, methods, inheritance. Biopython. Access to databases and simple searches.

BLOCK II. R.
Topic R1. Variables, expressions, operators in R. Types. Variables. Operators and precedence. Expressions. Data structures. Vectors. Type of data. Categorical data.

Topic R2. Conditional execution, recursion, iteration in R. Logical operators. Execution flow (if). Conditional and alternative execution. recursion. Iteration (for, while).

Topic R3. Reading and writing data into and from R. Directory tree structure. Reading files from disk. Parsing files. File writing. Handling of arguments passed by command line. Organization of data in the form of a “Data Matrix”.

Topic R4 Functions and modularity in R. Functions: arguments, return values, void functions. Composition of functions. Local and global variables. Import libraries. CRAN and Bioconductor.

Topic R5. Data visualization in R. "Base" system of graphics in R. Selection of data visualisations. Basic chart parameters: color, labels, axes, legend.

Topic R6. Data transformation in R. Basic transformations: sort, merge, cbind, subset. Advanced transformations: family apply, aggregate.

Topic R7. Project development in R. Generation of reports with markdown.

Topic R8. Advanced data visualization with ggplot. Introduction to ggplot ("grammar of graphics plot"). Syntax in ggplot2. Interactive graphics. Charts by categories (facets). Topic 14. Applications of R specific to biochemistry and molecular biology. Analysis of "-omics" data.

Topic R9. R for specific Molecular biology applications: -omics data analysis.

1.14. Course bibliography

-

2. Teaching-and-learning methodologies and student workload

2.1. Contact hours

	#horas
Contact hours (minimum 33%)	48.5
Independent study time	101.1

2.2. List of training activities

Activity	# hours
Lectures	
Seminars	
Practical sessions	
Clinical sessions	
Computer lab	41
Laboratory	
Work placement	
Supervised study	
Tutorials	2
Assessment activities	5
Other	

3. Evaluation procedures and weight of components in the final grade

3.1. Regular assessment

The course includes a partial exam and a final exam for each block of the course (python and R). Each student can decide whether to take only the partial, only the final or both exams in each one of the blocks. The grade of the block in "exam" section will be the maximum reached in the two exams (the exams not presented will be counted with a value of zero).
In addition, the problems, practical cases of programming and comments to the programs presented by peers will be evaluated. The activities not carried out, or delivered after the deadline, will be counted with a value of zero in the calculation of grades.

The final course grade is calculated as follows
- Resolution of python exercises and practical programming cases: 15%.

- Exam block I (python): 30% calculated as a maximum (partial_python, final_python)
- R Project: 30% (Both final and intermediate prototypes will be evaluated)
- Exam block II (R): 15% calculated as a maximum (partial_R, final_R)
- Comments/evaluation of practical cases (python and R): 10%.

3.1.1. List of evaluation activities

Evaluatory activity	%
Final exam	35
Continuous assessment	65

3.2. Resit

Students qualified as "Not evaluated" and "Failed" may choose to be evaluated in a extraordinary call by taking a single exam. In this case, the course grade will be the weighted average of the grade obtained in the extraordinary exam (with a weight of 60%) and the grade obtained in the exercises and practical programming cases (weight 40%). Before the final extraordinary exam, the students will have the opportunity to complete the exercises and practical programming cases not handed in during the course.

3.2.1. List of evaluation activities

Evaluatory activity	%
Final exam	60
Continuous assessment	40

4. Proposed workplan

-