UNIVERSITÀ
DEGLI STUDI DI MILANO-BICOCCA

COURSE SYLLABUS

Introduction To Bioinformatics

2526-3-E3101Q116

Aims

The student will know some fundamental problems and algorithms in bioinformatics. The student will be able to write small Python programs to solve some problems in bioinformatics, using also data originating from publicly available databases.

Knowledge and understanding

This course provides basic knowledge and understanding on:

- Algorithms on biological sequences
- Data structures to index biological sequencing
- Algorithms for phylogeny reconstruction
- Unix shell
- Python programming in bioinformatics
- File Formats used in bioinformatics

Ability to apply knowledge and understanding

At the end of the course the students will be able to:

- decide which algorithms and data structures can be used to solve some problems in bioinformatics
- write Python programs for bioinformatics problems
- write small pipelines that use Python and the Unix shell

Making judgements

At the end of the course, the students will be able to interpret the results of a bioinformatics analysis.

Communication skills

At the end of the course, the students will be able to communicate the results of a bioinformatics analysis.

Learning skills

At the end of the course, the students will have developed the ability to:

- learn new libraries and software tools in bioinformatics
- tackle new computational problems in genomics

Contents

Fundamental problems and algorithms in bioinformatics. Pattern matching.Sequence Alignment. DNA sequencing. Evolutionary histories. Managing biological data (standard formats in Bioinformatics) and genomic databases.

Detailed program

1. Pattern matching: Algorithms and Data Structures. Karp-Rabin algorithm, Dömölki algorithm.
2. Suffix trees and suffix arrays: management, pattern matching and applications to the longest substring problem.
3. Sequence Alignment of two strings. Global alignment, local alignment, band alignment. Linear and generic gap cost. Multiple sequence alignment.
4. DNA Sequencing. Overlap graphs and de Bruijn graphs.
5. Evolutionary trees. Character-based models. Gusfield's algorithm for the perfect phylogeny. Distance-based models: ultrametrics and additive distance. UPGMA and Neighbor Joining algorithms. Max likelihood.
6. Genotypes and haplotypes. Single individual and pedigree cases.
7. Bionformatics open source software development methodologies
8. Linux shell
9. Introduction to genomic data
10. Standard formats in Bioinformatics: FASTA, FASTQ, Gene Transfer Format (GTF), Sequence Alignment Map (SAM/BAM)
11. Introduction to Python
12. Pandas
13. Biopython

Prerequisites

Time and space complexity
Basic data structures: lists, arrays, search trees, dictionaries
Sorting algorithms: radix sort, merge sort
Memory hierarchy
Wiriting a short program, in any programming language

Teaching form

Lectures and Laboratory. The individual study can use the e-learning platform to enrich the standard activities and to self assess the level of competence acquired during the course.

All activities are in-person and will be neither recorded nor streamed. The teaching language of this course is Italian. The activities will be:

- 18 lectures, 2 hours each, with an initial part in unidirectional mode and a second part in interactive mode.
- 24 lab activities, 2 hours each, with an initial part in unidirectional mode and a second part in interactive mode.

Textbook and teaching resource

The adopted textbook is “Algorithms on Strings, Trees and Sequences”, by Daniel Gusfield, Cambridge Univ. Press. The library has some copies, also as ebook.

The book "An Introduction to Bioinformatics Algorithms" by N. Jones, P. Pevzner is used only for some parts on phylogeny reconstruction and on genome sequencing.

The books "Theoretical Evolutionary Genetics" by J. Felsenstein and Population and Quantitative Genetics by Graham Coop are used for some topics on phylogeny reconstruction and on haplotypes.

The book Think Python by A. B. Downey is used for introducing the Python language.

The Pandas library is covered in the book Python Data Science Handbook by VanderPlas.

Semester

Second semester

Assessment method

The assessment has a written exam and a project work.

The written exam is taken individually, on the algorithmic topics presented during the lectures. This part consists of open-ended questions. The written exam is 1 hour long and contains 4 questions. Of those questions, you have to answer to 3 of them.
The evaluation of the written exam is based on the correctness and completeness of the answers, and on the ability to identify the essential elements of a topic.

The project work consists of devloping a Python program by a single student. Students can choose the topic of the project among
alternatives proposed by the teacher at the end of the course.

The project will be discussed after the student has passed the written exam and within a year since the date of the written exam.

The final grade is obtained by weighting 50% of the degree of the written exam and 50% the project work, but you have to pass both parts. There are no in-progress written exams.
Beware that you must be registered via "segreterie online" to take the exam. If you are not registered, you will not allowed to take the exam. No exceptions will be made.

There will be no midterm assessments.

Office hours

Office hours with Professor Raffaella Rizzi will be held in person. Yoi can send an email to raffaella.rizzi@unimib.it.
Office hours with Professor Della Vedova will be held online. You can book a meeting at https://www.unimib.it/gianluca-della-vedova

Sustainable Development Goals

GOOD HEALTH AND WELL-BEING