CityUniversity of Seattle

Syllabus

SCHOOL OF TECHNOLOGY AND COMPUTING

CS 531: COMPUTATIONAL BIOLOGY

(3 Credit Hours)
Effective: January 2011

Access to the Internet is required.
All written assignments must be in Microsoft-Word-compatible formats.
See the library's APA Style Guide tutorial for a list of resources that can help you use APA style.

CS 531: COMPUTATIONAL BIOLOGY

FACULTY

Faculty Name:

Contact Information:

[Instructor may insert personal message if desired]

COURSE DESCRIPTION

The field of computational and systems biology represents an integration of concepts and ideas from the biological sciences, engineering disciplines, and computer science. Recent advances in biology, including the human genome project and massively parallel approaches to probing biological samples, have created new opportunities to understand biological problems from a systems perspective. This course offers a clear exposition of the algorithmic principles driving advances in bioinformatics. Accessible to students in both biology and computer science, it strikes a unique balance between rigorous mathematics and practical techniques, emphasizing the ideas underlying algorithms rather than offering a collection of apparently unrelated problems. This course introduces biological and algorithmic ideas together, linking issues in computer science to biology and thus capturing the interest of students in both subjects.

COURSE RESOURCES

Required and recommended resources to complete coursework and assignments are listed on the My.CityU portal at Library>Resources by Course.

CITYU LEARNING GOALS

This course supports the following City University learning goals:

- Professional Competency
- Strong Communication and Interpersonal Skills
- Critical Thinking

PROGRAM CONTEXT

This course provides the foundation for all Masters of Science in Computer Science (MSCS) program with School of Management.

The content of this course aligns with the following program outcomes:

- The ability to apply knowledge of computing and mathematics appropriate to the discipline
- An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
- An ability to function effectively on teams to accomplish a common goal
- An ability to communicate effectively with a range of audiences

COURSE OUTCOMES

In this course, learners will successfully demonstrate the following outcomes:

- Implement Motif Search algorithm in C#.
- Design and develop Exon chaining algorithm to examine a DNA sequence in C#.
- Implement protein sequence using graph algorithms using C#.
- Implement an algorithm for constructing keyword tree in C#.
- Using Clustering algorithm implement k-means clustering using C#.

CORE CONCEPTS, KNOWLEDGE, AND SKILLS

To achieve the goals of this course, you will need to understand the fundamentals of the following core concepts:

- Computing
- Computer Systems
- Mathematics
- Systems Development
- Program Development
- Software Engineering
- Visual Studio 2010
- C#
- Molecular Biology
- DNA mapping
- Brute Force
- Motifs
- Genome
- Statistical methods
- Gene Prediction
- DNA Sequencing
- Pattern Matching
- Clustering
- Hidden Markov

OVERVIEW OF COURSE GRADING

The grade you receive for the course will be derived using City University of Seattle's decimal grading system, based on the following:

Overview of Required Assignments	% of Final Grade
Motif Search Algorithm	15%
Exon Chaining Algorithm	15%
Protein Sequencing Using Graph Algorithm	20%
Combinatorial Pattern Matching Exercise	15%

Clustering Exercise	15%
Instructor determined assignments (also including participation)	20%
TOTAL	**100%**

SPECIFICS OF COURSE ASSIGNMENTS

Motif Search Algorithm

Motifs are the short sequences flanking the genes that are responsible for regulation of transcription when certain proteins, called transcription factors, bind to these motifs. Motif search is an important problem in biology. This problem in general requires finding short patterns of interest from voluminous data. Several variants of this motif search problem have been identified in literature. Input is n sequences of length m each. Input also is two integers' l and d. The problem is to understand a motif (i.e., a sequence) M of length l. It is given that each input sequence contains a variant of M. The variants of interest are sequences that are at a hamming distance of d from M. For the first assignment students will write Motif finding algorithm which discovers such motifs without any prior knowledge of how the motifs look. Students will use C# to implement this algorithm. The students will upload all code to http://teavsrv.cityu.edu for review by the instructor.

Components	% of Grade
Design Strategy	25%
Code Implementation	50%
Presentation of the code	25%
Total	**100%**

Exon Chaining Algorithm

Students are introduced to gene prediction algorithm. The identification of regions of similarity (candidate regions of conserved symmetry), the detection of genome rearrangements such as transpositions and inversions, and exon prediction. Students will implement the algorithm of examining a DNA sequence and determining which part of that sequence are genes. The algorithm can be a "greedy" or "non-greedy" algorithm but must be efficient in the time it takes to produce an output. Students will then write test case to test the C# algorithm. The students will upload all code to http://teavsrv.cityu.edu for review by the instructor.

Components	% of Grade
Design strategy	25%
Code Implementation	50%
Presentation of the code	25%
Total	**100%**

Protein Sequencing Using Graph Algorithm

The determination of protein sequences is a vital component to medical diagnostics and functional biological analysis. The technology used to sequence proteins is substantially more complex than the

technology used to sequence DNA. Efficient dynamic programming algorithms are available for a broad class of protein and DNA sequence comparison problems. The student can chose a dynamic or static programming algorithm for their proteien sequences programming assignment. Students will implement the algorithm to construct the spectrum graph given the sequence of a protein.Students will then write test case to test the C# algorithm. The students will upload all code to http://teavsrv.cityu.edu for review by the instructor.

Components	% of Grade
Design strategy	25%
Code Implementation	50%
Presentation of the code	25%
Total	**100%**

Combinatorial Pattern Matching Exercise

Biologists are often interested in finding matches of short sequences against entire genomes. Such as the TATAAAA motif found in the promoter region of most eukaryotic genes. Being able to find patterns within very large and sophisticated data structures can greatly assist in this task. In this exercise, students will implement an algorithm for constructing a keyword tree that identifies motifs and transcription factor binding sites in DNA sequences and searching those for recurrent structural motifs. Students will then write test case to test the C# algorithm that they have developed for this assignment. The students will upload all code to http://teavsrv.cityu.edu for review by the instructor.

Components	% of Grade
Design strategy	25%
Code Implementation	50%
Presentation of the code	25%
Total	**100%**

Clustering Exercise

Many of the new biological experimental techniques generate vast amounts of data. When viewed as a whole these data can be perplexing, however they are more sensible when alike data is clustered into groups. Mofif cluster is when the intial motif consists of four or more clusters that share similarities or properties and can be overlooked in the process of sequencing very large data structures of similar data. In this exercise, students will implement k-means clustering algorithm using C#. K-means is one of the most common clustinering algorithms. Students will then write test case to test the C# algorithm. The students will upload all code to http://teavsrv.cityu.edu for review by the instructor.

Components	% of Grade
Design Strategy	25%
Code Implementation	50%
Presentation of the code	25%
Total	**100%**

COURSE POLICIES

Late Assignments

Students are expected to meet submission requirements for assignments in a timely manner. Evaluation includes an assessment of timeliness. Late assignments jeopardize your learning, and may also penalize your classmates as most assignments will not be returned to students until all students have submitted their work. Late submission of assignments may be penalized up to 50% of the grade per week. Your instructor will provide additional details.

Quizzes, exams, and comprehensive assessments must be taken at the scheduled times. Any absences or late submissions must be approved by your instructor before the scheduled assessment date. Not completing a quiz, exam, or comprehensive assessment in a timely manner will result in a grade of zero unless a student has been preapproved by the instructor to complete the assessment at an alternative time.

Participation

Whether in class, online, or in a mixed mode setting, students will be graded on their participation in classroom discussions; their ability to present, explain, or defend alternative viewpoints; and the degree to which they have mastered the concepts and principles addressed in this course. Written work will be assessed not only on relevance to the subject presented, but also on adherence to good written form and professional presentation.

Students are expected to be actively engaged in all discussions as well as other activities. Active engagement means contributing substantive, thoughtful and reflective responses. For online classes, students must post their initial responses during the first three days of the week, and their responses to other students' postings during the last four days of the week

Professional Writing

Assignments require error-free writing that uses Standard English conventions and logical flow of organization to address topics clearly, completely, and concisely. CityU requires the use of APA style.

UNIVERSITY POLICIES

You are responsible for understanding and adhering to all of City University of Seattle's academic policies. The most current versions of these policies can be found in the University Catalog that is linked from the CityU Web site.

Scholastic Honesty

Scholastic honesty in students requires the pursuit of scholarly activity that is free from fraud, deception and unauthorized collaboration with other individuals. You are responsible for understanding CityU's policy on scholastic honesty and adhering to its standards in meeting all course requirements. A complete copy of this policy can be found in the University Catalog in the section titled *Scholastic Honesty* under *Student Rights & Responsibilities*.

Attendance

Students taking courses in any format at the University are expected to be diligent in their studies and to attend class regularly.

Regular class attendance is important in achieving learning outcomes in the course and may be a valid consideration in determining the final grade. For classes where a physical presence is required, a student has attended if s/he is present at any time during the class session. For online classes, a student has attended if s/he has posted or submitted an assignment. A complete copy of this policy can be found in the University Catalog in the section titled *Attendance Policy for Mixed Mode, Online and Correspondence Courses*.

SUPPORT SERVICES

Disability Resources

If you are a student with a disability and you require an accommodation, please contact the Disability Resource Office as soon as possible. For additional information, please see the section in the University Catalog titled *Students with Special Needs* under *Student Rights & Responsibilities*.

Library Services

In order to help you succeed in this course, you have access to library services and resources 24 hours a day, seven days a week. CityU librarians can help you formulate search strategies and locate materials that are relevant to your coursework. For help, contact a CityU librarian through the *Ask a Librarian* service. To find library resources, click on the Library link in the My.CityU portal.

Smarthinking

As a CityU student, you have access to 10 free hours of online tutoring offered through Smarthinking, including writing support, from certified tutors 24 hours a day, seven days a week. Contact CityU's Student Support Center at info@cityu.edu to request your user name and password.