BCB410 – ORGANIZATION AND SYLLABUS

1. Students

As a (required) course in the BCB curriculum, this course assumes the prerequisites and goals of fourth-year students in the BCB Specialist Program. Other students may participate but they may need to catch up on prerequisites in computer science or life-science courses that BCB students have taken at this point. They may also need to consider whether their objectives match the course objectives well.

Name	**eMail**
I	@gmail.com
J	@mail.utoronto.ca

2. Objectives

We have included an "Applied Bioinformatics" course in the curriculum to ensure that our students know enough about application issues around principles and algorithms of bioinformatics and computational biology to be able to put this into practice in a research lab setting. This is to support the Specialist Program goal: to prepare you for graduate studies in the discipline.

3. Organization

3.1. A syllabus of learning units

In the syllabus we identify learning units that are of the greatest interest and greatest relevance for the students in the class. We will go through the list and select the most suitable topics. Every student in class will take responsibility for one of the units.

3.2. Unit contents and delivery

The detailed contents for each unit should be discussed with me. In principle I expect you to lead a two hour session on your topic. Your tasks include[1]:

- **outline** of your unit contents, to be completed at least three weeks in advance; This is to include:

[1] Details may vary as required, by mutual agreement.

    - a detailed lecture outline that includes an introduction, discussion of algorithms, presentation of examples, exposition of practical- and implementation issues and an outlook on future developments in the field;
    - suitable pre-reading material;
    - an outline of excercises for the class;
- **iteration of your unit contents** with the coordinator, to be completed at least two weeks in advance.
- **developing a set of exercises** (iterated with the coordinator) around the implementation of your topics[2], at least one week in advance;
- **communication of pre-reading materials** to your classmates, at least one week in advance;
- **delivery of your lecture** at a sufficiently technical level to be appropriate for an advanced fourth-year course;
- **communication of exercise materials** to your classmates, at or directly after your lecture;
- **drafting a final-exam question** that tests the successful completion of the exercises, at the latest one week after your lecture.

Students' tasks include:

- pre-reading before class;
- active participation in the discussion;
- feedback on the exercises and completion in due time.

3.3. Schedule

The class schedule will be variable and tailored to your needs and the course contents. If nothing else is determined, lectures wil default to Wednesdays, 10:00 to 12:00.

3.4. Marking

• Design and coordination of your unit	20%
• Delivery and contents of presentation	20%
• Quality of exercises/assignments	30%
• Participation	10%
• Final exam	20%

3.5. Discussions and contact

We will install a Google group for contact and discussions.
We will use Dropbox to share files.

[2] These are to be iterated with me, they may include installation of tools, programming tasks, analysis tasks, synthetic or biological sample data etc.

4. Learning Unit options

Details of the learning units are to be coordinated with the coordinator.

4.1. Tools and skills

4.1.1. UNIX tools

Date: –
Contents: automating and scripting, Perl one-line commands

4.1.2. UNIX system security

Date: –
Contents: state-of-the art for a current research lab; update of lab systems.

4.1.3. Cytoscape

Date: –
Delivered by: N.N
Contents: Advanced use

4.1.4. MySQL and MariaDB

Date: –
Delivered by: N.N
Contents: Practice (and best practice) of deployment

4.1.5. HMMER

Date: Friday, 4 November **X**
Delivered by: Joe, 2011
Contents: Theory of profile HMMs and practice of HMMER 3 deployment and use.

4.1.6. Data integration - BIOMART

Contents: Biodata integration, and data-mining of complex, related, descriptive data.

4.2. General Algorithms

4.2.1. Pattern discovery in sequences, structures and networks

Date: Thursday, 10 November **X**
Delivered by: Omar, 2011
Contents: Footprinter, Gibbs sampling, MEME

4.2.2. Correlation analysis in large datasets

Date: (*not scheduled for this term*) –
Contents: Covariance matrices and their interpretation, application to large problems, collaborative filtering

4.2.3. Modern clustering approaches

Date: (*not scheduled for this term*) –
Contents: Algorithms and choice (including graphical methods (MCL), flow based methods (RRW) and spectral methods)

4.2.4. Cluster metrics

Date: (*not scheduled for this term*) –
Contents: Cluster quality metrics (Akaike, BIC)–when and how

4.2.5. The Map equation

Date: (*not scheduled for this term*) –
Contents: A network flow approach to hierarchical partitioning of large datasets

4.2.6. Support Vector Machines

Date: –
Contents: Implementations and aplication to biological data

4.3. Algorithms on Sequences

4.3.1. Dynamic Programming and Optimal Pairwise Sequence Alignment

Date: **X**
Delivered by: Zahra, 2011
Contents: appropriate problems for DP, procedural and recursive formulation of solutions

4.3.2. Multiple Sequence Alignment

Date: **X**
Delivered by: Nirvana, 2011
Contents: Modern MSA algorithms, principles and performance

4.3.3. Assembly, genomes, long- and short-reads

Date: **X**
Delivered by: Christian, 2011

Contents: Challenges and algorithms

4.4. Algorithms on Structures

4.4.1. Protein ligand docking

Date: **X**
Delivered by: Shirin, 2011
Contents: Geometric hashing approaches to substructure (subgraph) matching

4.5. Algorithms on Trees

4.5.1. Bayesian approaches for phylogenetic trees, tree comparison

Date: –
Contents:

4.6. Algorithms on Networks

4.6.1. Non-trivial network metrics

Date: –
Contents: Centrality metrics, other metrics on topology, small-world- vs. random-geometric controversy

4.7. Web technologies

4.7.1. HTML5

Date: Friday, 2 December **X**
Delivered by: Garon, 2011
Contents: Principles and coding examples

4.7.2. PHP

Contents: PHP

4.7.3. SADI Semantic Automated Discovery and Integration

Date: Friday, 18 November **X**
Delivered by: Huajing, 2011
Contents: Data and service integration through standard interfaces

4.8. Statistics

4.8.1. R programming

Date: Friday, 25 November **X**
Delivered by: Martin, 2011
Contents: Defining function and programs in R

4.8.2. Linear and non-linear regression in R

Date: –
Delivered by: N.N
Contents: Calculation, confidence limits and interpretation

4.8.3. Hypothesis testing in R

Date: –
Delivered by: N.N
Contents: Major approaches and when to apply them (including simulation methods for arbitrary PDFs)

4.8.4. Clustering and density estimation in R

Date: –
Delivered by: N.N
Contents: Partitioning and hierarchical approaches

4.9. Graphics

4.9.1. PNG and SVG output

Date: –
Delivered by: N.N
Contents: Working with pixel-based and vector-based graphical output