# Assignment 2

## Due: March 2, 2010, in class

Scoring:

Undergraduate students:

**Question 1** (100 % of the total assignment score)

- implementation 50%
- optimal scoring matrix and the exons information: 25%
- output alignment: 25%

Any additional question: the corresponding bonus

Graduate students:

**Question 1** (70 % of the total assignment score)

- implementation 30%
- optimal scoring matrix and the exons information: 20%
- output alignment: 20%

**Question 2** (30 % of the total assignment score)

One of 2 variants: 30%

An additional variant: 30% bonus

Implementation: 30% bonus

**Question 1.** Align c-DNA (protein coding sequence), consisting of several exons to a larger region of genomic DNA. The c-DNA encodes for Mouse tumor protein (gi|223460748:50-868 Mus musculus tumor) and is supplied in the file *"CodingSequence.txt"*. The genomic sequence is a cut from Mouse chromosome 4, between positions 8256000 and 8258450, and is supplied in file "*Chromosome4_8256000_8258450.txt*".

Implement, in a language of your choice, a pairwise global alignment algorithm with affine gap penalties (see page 184 of a textbook or slides 38-43 of Lecture 7) which takes as an input 2 sequences and a scoring matrix, performs dynamic programming and outputs a best alignment. A scoring matrix contains the cost for matches, mismatches and penalties for opening and extending gaps.

Play with the scoring schemes in order to align exons, which contain only several mismatches to their positions in the genomic DNA (Hint: for ideas see Lecture 8, slide 5).

**To be submitted:**

1. Code with the instructions of how to run it
2. An optimal scoring matrix, the number of discovered exons in the coding sequence and their start and end positions in the genomic sequence
3. A produced alignment in a separate file in form of a tab delimited 2-row texts:

```
ACCGA--TA
ACAGAATTA
```

## Question 2

### Variant A

Is it possible to develop a linear-space modification for the Miller-Myers algorithm to compute a longest common subsequence?

If you think that it is possible, then develop a linear-space modification for the Miller-Myers algorithm for computing a longest common subsequence. The original paper is in the file “A file comparison problem” which also contains the pseudocode of an original algorithm (the algorithm was presented in class, starting from slide 48 of Lecture 6). Hint: you may try an idea similar to reducing the space complexity in Hirshberg’s ‘divide and conquer’ approach. However, you may also need to think outside the box. An implementation of your idea is optional and will give you up to 30 bonus points.

If you think that it is impossible to modify the Miller-Myers algorithm in linear space, give a detailed explanation and justification of your answer.

### Variant B

If possible, develop a modification of Viterbi algorithm (p.393 of the book or slide 16 of Lecture 10), which runs in time $O(Q^2N)$, but uses only O(N) space. An implementation will give you 30 points bonus and is not required.

If you think that it is impossible to run the Viterbi algorithm in space less than O(QN), give detailed explanation and justification of your answer.