Lecturer: Shlomo Moran
TA: Ilan Gronau

- Exam Duration: 3 hours
- The Exam has 3 questions which sum up to 105 credits.
  The Maximum possible grade is 100.
- Material: Lectures and Tutorial slides,
  Books:  Durbin et al, Setubal et al, Gusfield

<table>
<thead>
<tr>
<th>/35</th>
<th>Question 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>/35</td>
<td>Question 2</td>
</tr>
<tr>
<td>/35</td>
<td>Question 3</td>
</tr>
<tr>
<td>/100</td>
<td>Total</td>
</tr>
</tbody>
</table>

Good Luck
Question 1 (35 points)

The "Gap Restricted Alignment problem", GRA, gets as input two sequences $S, T$ of lengths $m, n$ (respectively) over some alphabet $\Sigma$, an integer $k$, and an additive scoring scheme defined by a function $\sigma: (\Sigma \cup \{-\}) \times (\Sigma \cup \{-\}) \rightarrow \mathbb{R}$. It is required to find an alignment of $S$ and $T$ of maximum score, under the restriction that there are at most $k$ indels in the alignment (recall that an indel is "-" matched with a letter from $\Sigma$). In your proofs you may assume correctness of any of the algorithms taught in the lectures or tutorials.

1. (13 pts) For integers $i, j, l$, we denote by $S(i, j, l)$ the maximal score of an alignment of $S[1,...,i]$ with $T[1,...,j]$ which contain at most $l$ indels. Write a recursive formula for $S(i, j, l)$, including all termination conditions (boundary cases). Argue why your formula and termination conditions are correct.

2. (9 pts) Write a dynamic programming algorithm (in pseudo code) for solving the Gap Restricted Alignment problem. Analyze the complexity of your algorithm.

3. (13 pts) The Refined Gap Restricted Alignment problem, RGRA, is similar to GRA, except that instead of one integer $k$ the input contains two integers $k_s$ and $k_t$, and the output alignment is required to have at most $k_s$ "-"'s in $S$ and at most $k_t$ "-"'s in $T$. Write a recursive formula (including termination conditions) for a dynamic programming algorithm which solves RGRA, and analyze the complexity of the resulted algorithm. There is no need to explicitly write down the algorithm.
The DNA of species in planet Melmek contains two letters: A and B. In the evolutionary process in Melmek, a mutation of a letter occurs in two steps: first the letter may be deleted, and in case the letter is not deleted it can be changed to the 2nd letter. The unknown probabilities of these events are respectively $p_{del}, p_{A\rightarrow B}, p_{B\rightarrow A}$.

There are two species in Melmek, S (for Son) and its direct ancestor F (Father). Scientists were able to deduce that the DNA of F contained a sequence of two letters "AX", where "X" is equally likely to be A or B ($\text{Prob}(X=A) = \text{Prob}(X=B) = 0.5$).

1. **(12 pts)** Describe the probability space defined by the evolution of the two letters "AX" in F into a sequence (of up to two letters) in S:
   - Write down all the "simple events".
   - For each event write its probability (as a function of $p_{A\rightarrow B}, p_{B\rightarrow A}, p_{del}$), and its contribution to the six statistics used by the EM algorithm.

Later information showed that the sequence "AX" of F evolved into the sequence "A" in S.

2. **(5 pts)** For given $p_{A\rightarrow B}, p_{B\rightarrow A}, p_{del}$, write down the probability of the above scenario of evolution (of "AX" in F evolving into "A" in S).

3. **(10 pts)** Write a single round of the EM algorithm to estimate the parameters $p_{A\rightarrow B}, p_{B\rightarrow A}, p_{del}$ which maximize the likelihood of the above scenario, starting with arbitrary initial parameters $p_{A\rightarrow B} = x, p_{B\rightarrow A} = y, p_{del} = z$. Show that the outcome is independent on the initial parameters.

4. **(8 pts)** What are the parameters which maximize the likelihood of the above scenario? Justify your answer.
This question deals with Fitch's algorithm for maximum parsimony.

1. **(3 pts)** Define the specification of sets $R_v$ calculated by the bottom-up phase of Fitch's algorithm for every vertex $v$ of the phylogenetic tree (as defined in class).

The following questions relate to the tree, with assignment of DNA characters to the leaves described in the figure below.

2. **(3 pts)** Calculate sets $R_v$ for vertices 1-13. Explain your calculations.

3. **(9 pts)**
   a. What is the maximum-parsimony score?
   b. How many possible outputs can Fitch's algorithm yield on this input?
      Note: Fitch's algorithm outputs a state assignment to every internal node.
   c. Give an example of one possible output.

4. **(12 pts)** Prove or disprove (הוכח או הפרך) the following statement:

   "Fitch's algorithm yields all optimal solutions (i.e. maximum parsimony assignments) on the tree described in the above figure."

5. **(8 pts)** Prove or disprove (הוכח או הפרך) the following statement:

   "The tree described in the above figure provides an optimal solution to the big parsimony problem."