Computational Molecular Biology
Final Exam

LIACS Room 174
Tuesday June 9th 2015
14.30 – 17.30

- State your name and student number on every page of your answers.
- Every assignment has the same weight. There are 12 assignments.
- Always fully explain your answers,
- Please note that you have a total of 3 hours to answer the questions.
- It is a closed book exam, no books, notes, smart phones, etc. allowed.

1. Calculate the score of the DNA sequence alignment shown below using the following scoring rules: +3 for a match, -2 for a mismatch, -3 for opening a gap, and -1 for each position in the gap.

```
A A C A C C G T G A A C T C A C A - - - C
        |   | | | |           | |       |
- - - - C - G T G A G T C A G C A T A A C
```

2. Given two DNA sequences S and T of length N and M, respectively. Which algorithm can be used to find an optimal local alignment of these two sequences? What is the best space-, and time-complexity of the algorithm you proposed? How does this compare to the best heuristic algorithms that solve this problem?

3. Describe how Hidden Markov Models can be used to find the optimal alignment for a set of sequences?

4. When is the MAQ algorithm used? What are the important characteristics of the MAQ algorithm? Depict templates that will be able to handle 2 mismatches.

5. In the BLAST algorithm a threshold T is used that determines which w-length (typically w~12 for DNA sequences) substrings of database sequences have an alignment score with words from the query string that are high enough. T can be varied. Describe the impact of varying T on the result and the time complexity of BLAST.

6. Algorithms that solve the multiple alignment problem often use a special score, for example the Sum-of-Pairs Score.
    1) What is the main drawback of the sum-of-pairs score?
    2) Give the definition of the sum-of-pairs score.
    3) Give pseudo code of an algorithm that solves the multiple alignment problem using the sum-of-pairs score.

7. The architecture of single-sequence RNA secondary structure prediction algorithms can be expressed using context free grammars. The *Nussinov* grammar has the following production rules:
    1) S → S r
    2) S → S r S r’
    3) S → e

    Where r, and r’ are terminals from the set {a, c, u, g}, and S a non-terminal, and ‘e’ the empty string.
    Explain each of the production rules in terms of the secondary structures it produces. Give the sequence of rules for the following secondary structure (((-))) (bracket notation) of the sequence ‘cagccug’. What is the problem with this ‘predicted’ secondary structure? How would you extend your *Nussinov* grammar to solve this problem?

8. The Forward Algorithm is an important algorithm when working with HMMs. What does it compute? Mention 2 important applications. What is its complexity?

9. Draw a keyword tree with failure links for the following set of words: {sequel, sense, quant, anti}.

10. For scoring of an alignment of two sequences often a special gap function is used. Give the definition of the affine gap scoring function. What impact has an affine gap function on the time and space complexity of a global sequence alignment algorithm such as Needleman-Wunsch’s algorithm?

11. Draw a 2-dimensional De Bruijn Graph for the following reads: ACTAC, CACCA, CACCC, ACCCA, TACAC.

12. Let T be a given text. The Burrow-Wheeler Transform of T is denoted by BWT(T). Assume that for T, BWT(T) = ‘c$agcgtcgcta’. Determine the original text T using the UNPERMUTE algorithm. Note: ‘$’ is the lexicographical smallest symbol at the end of the original text T.