Stockholm University
Department of Mathematics
Examiner: Marc Hellmuth

2. Homework "DA7065 Computational Biology"

Exercises that are not *marked with ⋆ are for all participants. Exercises marked with ⋆ are intended as additional challenges for PhD students. However, all students are welcome to attempt solving ⋆-exercises to earn extra points.*

Exercise 1: 10+2.5+2.5=15p

(a) Implement the Simple Linear-Time Exact Matching Algorithm that uses the Z-Algorithm (see online-script page 21).
Hand in your (well-commented) source code. Document how to compile and run your program.

(b) Use your implementation to find the positions and number of occurences of the pattern
$p =$ *"**schon an die Lippen**"* and $p =$ *"**Nacht**"*.
in "Faust" by Goethe (p without quotation marks).

(c) Use your implementation to find the number of occurences of the pattern $p =$ `gcgg` in the genom of the gut bacteria *E. coli*.

"Faust" and the *E. coli* genom is provided at the kurser-homepage (below EXERCISES, Exercise 2, working-material).

Exercise 2: 5+1+2+2 =10p
Given the strings $u =$`GTTTAAG` $v =$`GAAGA` and cost function

$$\delta(a,b) = \begin{cases} 0 & \text{if } a = b \\ 1 & \text{if } a \neq b \text{ and } a, b \neq - \\ 2 & \text{else} \end{cases}$$

(a) Compute the cost matrix D for the strings u and v.
(b) What is the optimal alignment score?
(c) Give one possible optimal alignment for u and v.
(d) How many optimal alignments are there for u and v - Explain shortly your results.

Exercise 3: 3+4.5=7.5p
Use the RNAfold WebServer⋆ to compute the MFE-structure of the sequences

$$s1 = \texttt{AAAUGCGGUUCCAAGUAACC}$$
$$s2 = \texttt{CCAAUGAACCUUGGCGUAAA}$$
$$s3 = \texttt{ACGUACGUACGUACGUACGU}$$

(a) Give the respective MFE-structures $S1$, $S2$ and $S3$ for the sequences $s1$, $s2$ and $s3$ in *bracket-notation*.

(b) Prove or disprove: There is a sequence $s \in \{A, C, G, U\}^{20}$ that realizes all three MFE-structures. Give such a sequence if one exists.

⋆ `http://rna.tbi.univie.ac.at/cgi-bin/RNAWebSuite/RNAfold.cgi`
Use the options: *minimum free energy (MFE) only* and *avoid isolated base pairs*

Exercise 4: 7.5p
Let $S(n,k)$ denote the number of possible secondary structures ($\Theta = 1$) of size n having exactly k basepairs. Let $S(n,0) = 1$ for all n and $S(n,k) = 0$ for $k \geq n/2$.
Show that for all $n \geq 2$ holds:

$$S(n+1,k+1) = S(n,k+1) + \sum_{j=1}^{n-1} \left[\sum_{i=0}^{k} S(j-1,i) S(n-j,k-i) \right].$$

Exercise 5⋆: 15p

Implement a program in your favorite programming language that takes as input two sequences: the first sequence is a bracket-dot expression consisting of characters '`(`', '`)`' and '`.`' and the second sequence consists of the characters '`A`', '`C`', '`G`', or '`U`'

The program should accomplish the following tasks:

(a) *Verify that the provided bracket-dot expression "represents" a* valid *RNA secondary structure*

Here, we say that an RNA secondary structure is *valid* if the following conditions are satisfied:

(i) The bracket-dot expression represents a secondary structure according to the definition on slide 7 (5-RNA-slides.pdf), where $\Theta = 0$ (the number of minimum unpaired positions enclosed by a base-pair), i.e., Θ can be ignored.

(ii) Each hairpin loop has length at least three.
To recap, a hairpin loop is a series of unpaired bases that are closed by a base pair. For example, the secondary structure `(.(....).)` has a single hairpin loop of length 4. So, the structure `((((..))))` is not valid because it has a hairpin loop of only two bases. The latter assumption is justified by sterics.

The program should return the list of base-pair-positions in case we have a valid RNA secondary structure and otherwise `false`

(b) *Checks if the given sequence realizes the given secondary structure in case the secondary structure is* valid.

The program should return `True` if the sequence realizes the secondary structure and `False` otherwise. For the latter task, add an option that allows/forbids "wobble base-pairs" `G-U`, `U-G`.

Provide the well-documented source code and also instruction on how to compile or execute your program. Test your program on the following instances and provide the output of your program where in all but the sequence specified by '⋆' wobble base-pairs are allowed.

```
GCAUCUAUGC
(((....)))

GCAUCUAUGU
(((....)))


GCAUCUAUGU
(.(....).)


AUUGAUGCACGUGCAUCCCCAGCGGGUCCCGCGAGCUCACCCCCUUCCAAAAGCACCACGUGCCAGGCCUCGCCCCCGGAAGUAUACCUGUGAGCCAGA
...(((((....)))))....((((...))))..((((((...(((((....((((...))))..(((...)))...)))))).......))))))....


GCAUCUACGC
(((....)))


GCAUCUAUGU*
(((....)))


GCAUCUAUGU
(.(....)).


GCCCUUGGCA
(.((..))).


GCCCCCCC
(.)...


AUUGAUGCACGUGCAUCCCCAGCGGGUCCCGCGAGCCCACCCCCUUCCAAAAGCACCACGUGCCAGGCCUCGCCCCCGGAAGUAUACCUGUGAGCCAGA
...(((((....)))))....((((...))))..((((((...(((((....((((...))))..(((...)))...)))))).......))))))....
```

Deadline: Wednesday - Feb 21