Note: Any solution to an algorithm design question MUST contain the following

four sections:

(1) Problem statement. A clear unambiguous statement of the problem to

be solved, which includes the input, the output, and the object function

with the constraints.

(2) Algorithm description. A clear, unambiguous description of the algo-

rithm.

(3) Correctness proof. A convincing mathematical argument that the algo-

rithm described solves the computational problem described.

(4) Time analysis. A time analysis of the algorithm, up to order, in terms of

all relevant parameters.

You may use any algorithms and data structures from class.

1. RNA Base Pair Maximization Problem (60 pts)

Formalize the pseudocode to predict an RNA secondary structure based on

the base pair maximization model including the trace back procedure for the pre-

dicted secondary structure. Please refer to slides 18-21 in 3.2.RNAfolding.pdf".

The secondary structure can be represented by the matching parentheses and dots

to denote paired and free bases, respectively. For example:

GAGCCAUUAGCUCAGUUGGUAGAGCAUCUGACUUUUAAUCAGAGGGUCGAAGGUUCGAGUCCUUCAUGGCUCA

(((((((..<<<<........>>>>..<<<<<.......>>>>>.....<<<<<.......>>>>>)))))).

\*USE zuker-sankoff\* actually I need to ask professor about this because I feel we need to use nussinov way

2. K-exons Spliced Alignment (40 pts)

One disadvantage of the exon chaining formulation of the spliced alignment algo-

rithm is that it may prefer to concatenate many short putative exons to maximize

the alignment score. Modify the spliced alignment algorithm to consider the opti-

mal alignment between a genomic sequence and an mRNA sequence with at most

k exons.

Spliced alignment recurrence

If i is not the starting vertex of block B:

• S(i, j, B) =

max { S(i – 1, j, B) – indel penalty

S(i, j – 1, B) – indel penalty

S(i – 1, j – 1, B) + δ(gi, tj) }

If i is the starting vertex of block B:

• S(i, j, B) =

max { S(i, j – 1, B) – indel penalty

maxall blocks B’ preceding block B S(end(B’), j, B’) – indel penalty

maxall blocks B’ preceding block B S(end(B’), j – 1, B’) + δ(gi, tj)

}

Spliced Alignment Solution

• After computing the three-dimensional

table S(i, j, B), the score of the optimal

spliced alignment is:

maxall blocks BS(end(B), length(T), B)