UPGMA tree inference

Problem 1.

					inferred nodes		
	L2	L3	L4	L5			
L1	14	14	14	8			
	L2	10	4	14			
		L3	10	14			
			L4	14			
				L5			

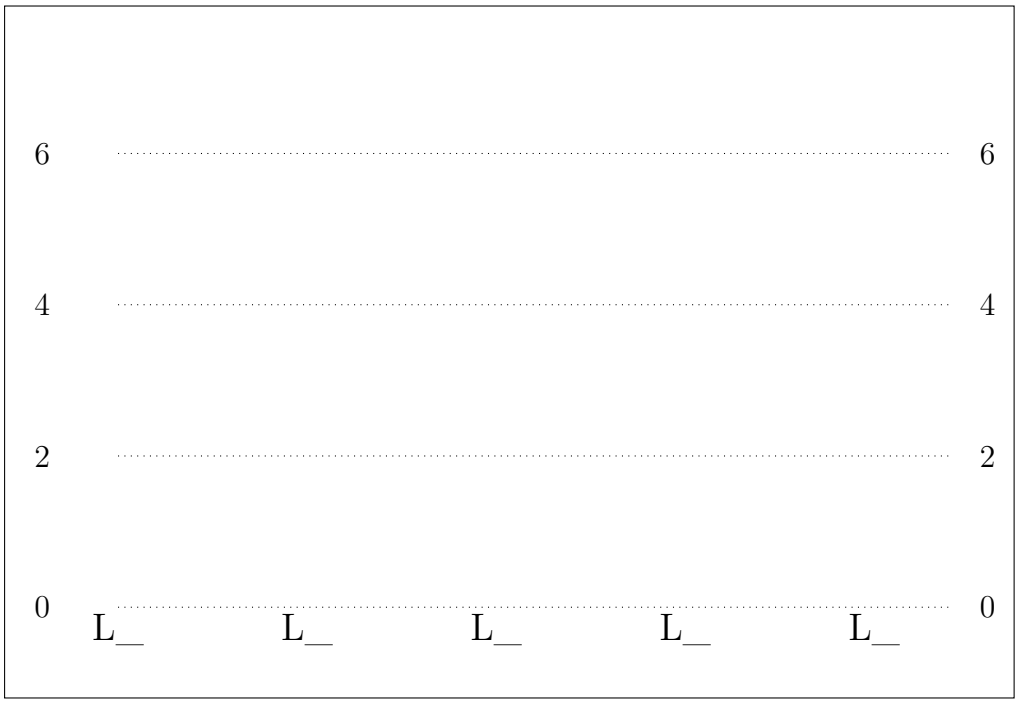
Problem 1a. Fill in all relevant distances computed by UPGMA.
Since this is a closed book test; let me remind you that UPGMA infers rooted trees assuming a constant speed of evolution.

The above matrix shows distances between species L1 to L5. Use UPGMA (Unweighted Pair Group Method Arithmetic averages) to infer a phylogenic tree. Indicate merged nodes in the blue boxes, so for example write "23" to indicate the inferred ancestor node obtained by merging L2 with L3, and "1:23" do indicate the node obtained by it with L1.

UPGMA tree inference

Problem 1**b**. Sketch the UPGMA inferred tree in box below.

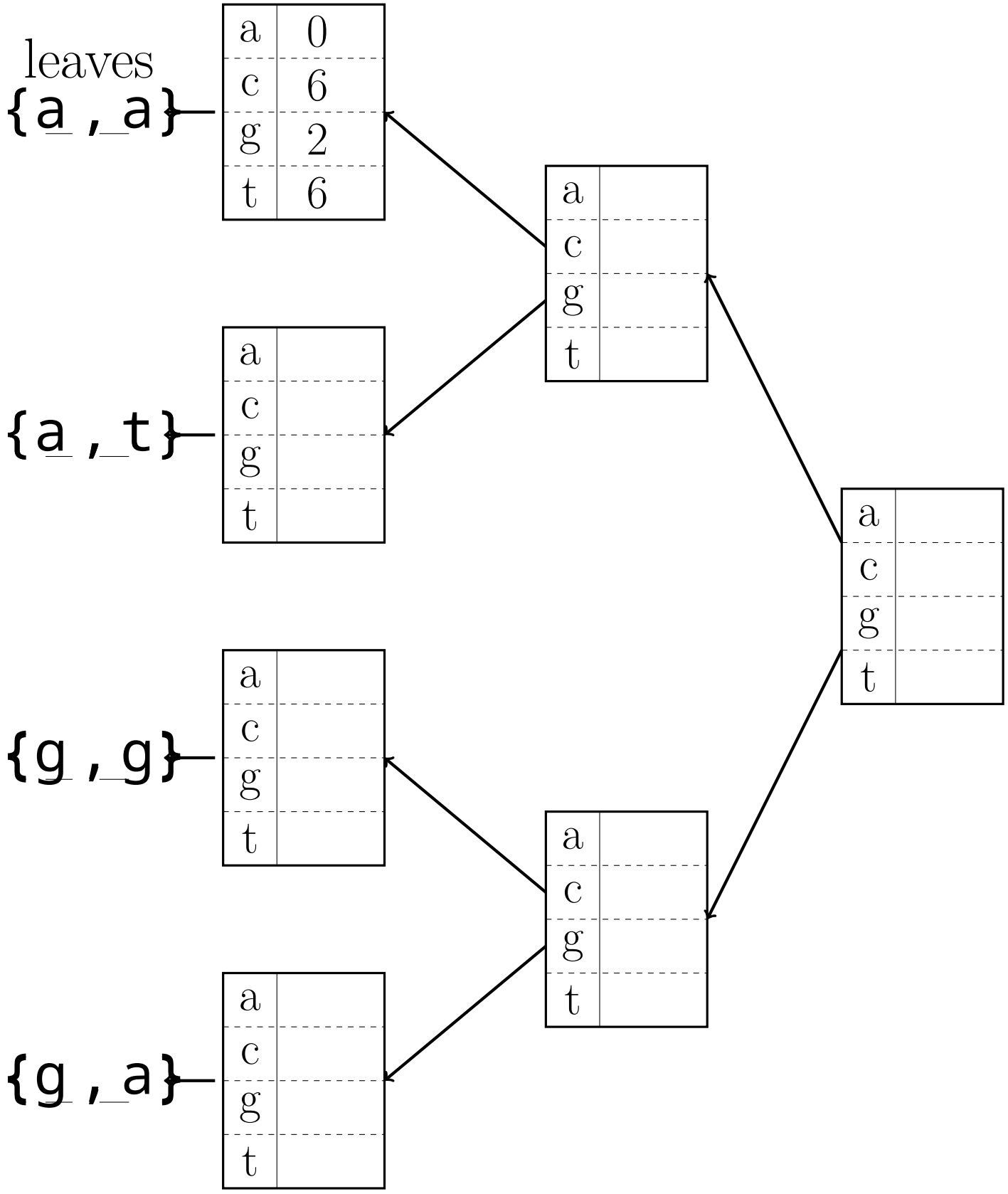
Sketch the tree inferred by UPGMA, with the leaves at height 0 and inferred ancestor nodes at the appropriate height. Indicate leaves by filling in "L1", "L2" etc. where there is a "L_". Show the root node as "root".



Problem 1**c**. Is this UPGMA tree reliable? State your reasons.

Weighted Parsimony

Problem 2. Fill in the minimal costs for assuming each nucleotide in the blanks below.



Problem 3.

		C	G	T	A	T	T	G
C								
G								
T								
C								
G								

According to instructions on the following page.

Problem 3**a**. Fill in this dynamic programming table.

Note that some scores could be better (higher) if adjacent alternating gaps were allowed. For example, cell representing aligning `CGT-` with `CG`.

Alternating adjacent gap alignment:

```
CGT-
C--G
*2 1
```

But we are forced to use:

```
CGT-
--CG
2 x1
```

Problem 3**b**. List any globally optimal alignments (here) and show traceback arrows on the dynamic table on the first page.

Scoring Parameters

Alignment scoring parameters: match score +2;
mismatch score transition -1, transversion -2.
Gap open score -3, gap extend score -1.

A transition is a substitution `a` $\longleftrightarrow$ `g` or `c` $\longleftrightarrow$ `t`, other single nucleotide substitutions are transversions.

For example, the alignment:

```
CGTTTTAAG
CGTCA---G
***xX 3 *
```

With 4 matches, 1 transition, 1 transversion, 1 gap opening, & 2 gap extensions, would score: $(4)(2) - 1 - 2 - 3 - (1)(2) = 0$

Problem 3**c**. (On the next page) List the recursive relationships needed to efficiently compute a minimal cost global alignment with affine gap costs. you may assume the scoring parameters penalize gaps strongly enough that an optimal alignment never has alternating adjacent gaps.

In other words:

```
x-x
yy-
```

alignment like this never optimal

Notation

Let $x = x_1, \cdots, x_n$, $y = y_1, \cdots, y_m$ denote the sequences.
Note that the indices start from 1; so that 0 can be used as an index to represent the empty string "before" the start of a sequence.
Let $m(\mathsf{a}, \mathsf{b})$ indicate the score of aligning characters `a` and `b` together (a match when `a`=`b`, otherwise a mismatch), Let g_o indicate gap open, and g_e indicate gap extension costs.

Continued next page...

Affine Gap Alignment

Additional Notation (additional definitions needed to write the recurrences below.)

Base Cases

Recurrences

Global Alignment Score (how to get the global score from your definitions.)

Problem 4.

Background

This problem is about using stochastic context free grammar formalization to model a RNA sequence motif, including secondary structure information.

We want to model a stem-loop structure like this one:

```
5'  (((NNNN)))Yu   3'
```

Where paired `()` represent bases paired in an RNA stem. Assume the following percent probabilities:

```
Y          c:40      u:60
N    a:20 c:30 g:20 u:30
()   c=g:25 g=c:25 a=u:20 u=a:20 u=g:05 g=u:05
```

Where 40 means 40%, etc; and `c=g` indicates a cytosine base paired with a guanine base in a stem structure.

Continued next page...

Problem 4**a**. Write a stochastic context free grammar to generate sequences consistent with the sequence motif and probabilities described above. Your grammar can use the symbols Y and N in the same way as above; but do not use "`( )`", or "`=`" in your grammar; instead use alphanumerical symbols such as H or H_1 etc.

Continued next page...

Problem 4**b**. What is the probability the model would generate the following sequence?

`ggccuaggcuuu`

Show enough calculation to justify your answer.