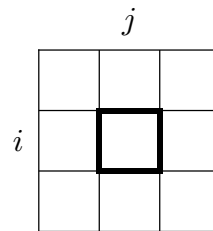


for how to compute $A(i, j)$ using the values in relevant neighboring cells, which as always with dynamic programming represent simpler sub-problems than $A(i, j)$.



Now, use your description to fill in all the cells in the chart below. Note that you will need to fill in base case cells “from scratch,” that is, not using the same number of cells as the general case. In addition to writing the score in each cell, you must draw an arrow or otherwise indicate which of the three possible decisions is made to produce the optimal score for the cell.

	T	A	A	G	G	T	C	A	-
A	7	8	10	12	13	15	16	18	20
A	6	6	8	10	11	13	14	16	18
C	4	5	6	8	9	11	12	14	16
A	6	4	4	6	7	9	11	12	14
G	7	5	3	4	5	7	9	10	12
T	9	7	5	3	3	5	7	9	10
T	8	8	6	4	3	3	5	7	8
A	10	8	7	5	3	2	3	5	6
C	12	10	8	6	4	2	1	3	4
C	14	12	10	8	6	4	2	1	2
-	16	14	12	10	8	6	4	2	0

Once the chart is filled in, write down the optimal way of aligning the two sequences, and verify that the penalty is correct.

Demonstrate (draw your own tidy chart) your dynamic programming algorithm for the instance of the sequence alignment problem with the sequences **GACATATTAC** and