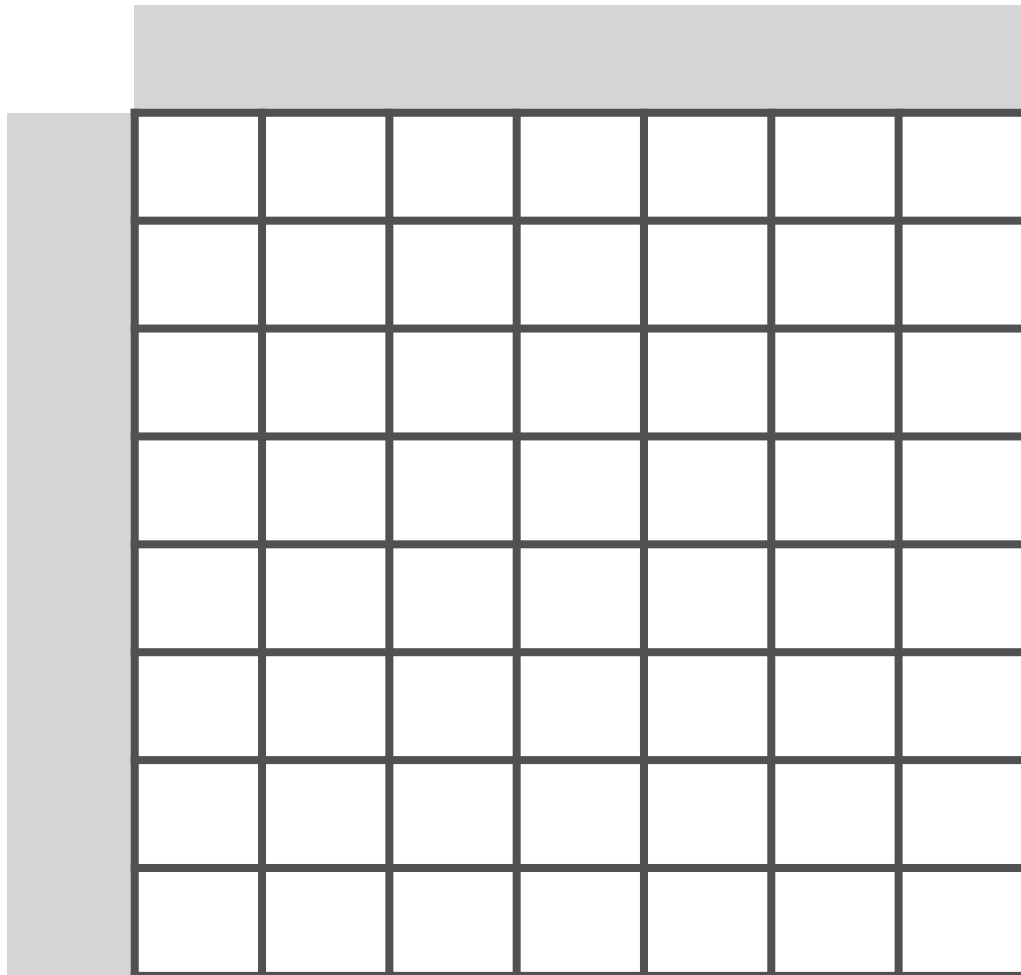


Find the **global alignment** between GATCGTT and TCTGAT with the  $\delta$  values from the table below. As a reminder the recursive formula for global alignment is given.

$\delta$	A	C	T	G	-
A	1.5	-1	-0.75	-1	-0.5
C	-1	1.5	-1	-0.75	-0.5
T	-0.75	-1	1.5	-1	-0.5
G	-1	-0.75	-1	1.5	-0.5
-	-0.5	-0.5	-0.5	-0.5	$-\infty$

$$V(i, j) = \max \begin{cases} V(i-1, j-1) + \delta(S[i], T[j]) & \text{match/mismatch} \\ V(i-1, j) + \delta(S[i], -) & \text{delete} \\ V(i, j-1) + \delta(-, T[j]) & \text{insert} \end{cases}$$



Best score: \_\_\_\_\_  
 Best alignment:

(0,0)	(1,0)	...	(n,0)
(0,1)	(1,1)	...	(n,1)
:	:		:
(0,m)	(1,m)	...	(n,m)

Find the **local alignment** between CACTUS and REACTSBAD with the  $\sigma$  values that follow the rules below. As a reminder the recursive formula for global alignment is given.

$$V(i, j) = \max \begin{cases} 0 & \text{align empty strings} \\ V(i-1, j-1) + \delta(S[i], T[j]) & \text{match/mismatch} \\ V(i-1, j) + \delta(S[i], -) & \text{delete} \\ V(i, j-1) + \delta(-, T[j]) & \text{insert} \end{cases}$$

$$\begin{aligned} \delta(-, x) &= -1 \text{ for } x \in \Sigma \\ \delta(x, -) &= -1 \text{ for } x \in \Sigma \\ \delta(x, y) &= 2 \text{ for } y = x \\ \delta(x, y) &= -1 \text{ for } y \neq x \end{aligned}$$


Best score: \_\_\_\_\_  
 Best alignment:

(0,0)	(1,0)	...	(n,0)
(0,1)	(1,1)	...	(n,1)
:	:		:
(0,m)	(1,m)	...	(n,m)

Find the **global alignment with affine gap costs** between AGGC and ATTGGGC with the scores. As a reminder the recursive formula for global alignment is given.

$$F(i, j) = \max \begin{cases} F(i-1, j) - b \\ G(i-1, j) - f_{a,b}(1) \end{cases}$$

$$f_{a,b}(i) = \alpha + \beta \cdot i$$

$$E(i, j) = \max \begin{cases} E(i, j-1) - b \\ G(i, j-1) - f_{a,b}(1) \end{cases}$$

$$G(i, j) = \max \begin{cases} G(i-1, j-1) + \delta(S[i], T[j]) \\ E(i, j) \\ F(i, j) \end{cases}$$

**F**


$\delta(x,y) = 5$  for  $y = x$   
 $\delta(x,y) = -1$  for  $y \neq x$   
 $\alpha = 2$   
 $\beta = 0.5$

Best score: \_\_\_\_\_  
 Best alignment: \_\_\_\_\_

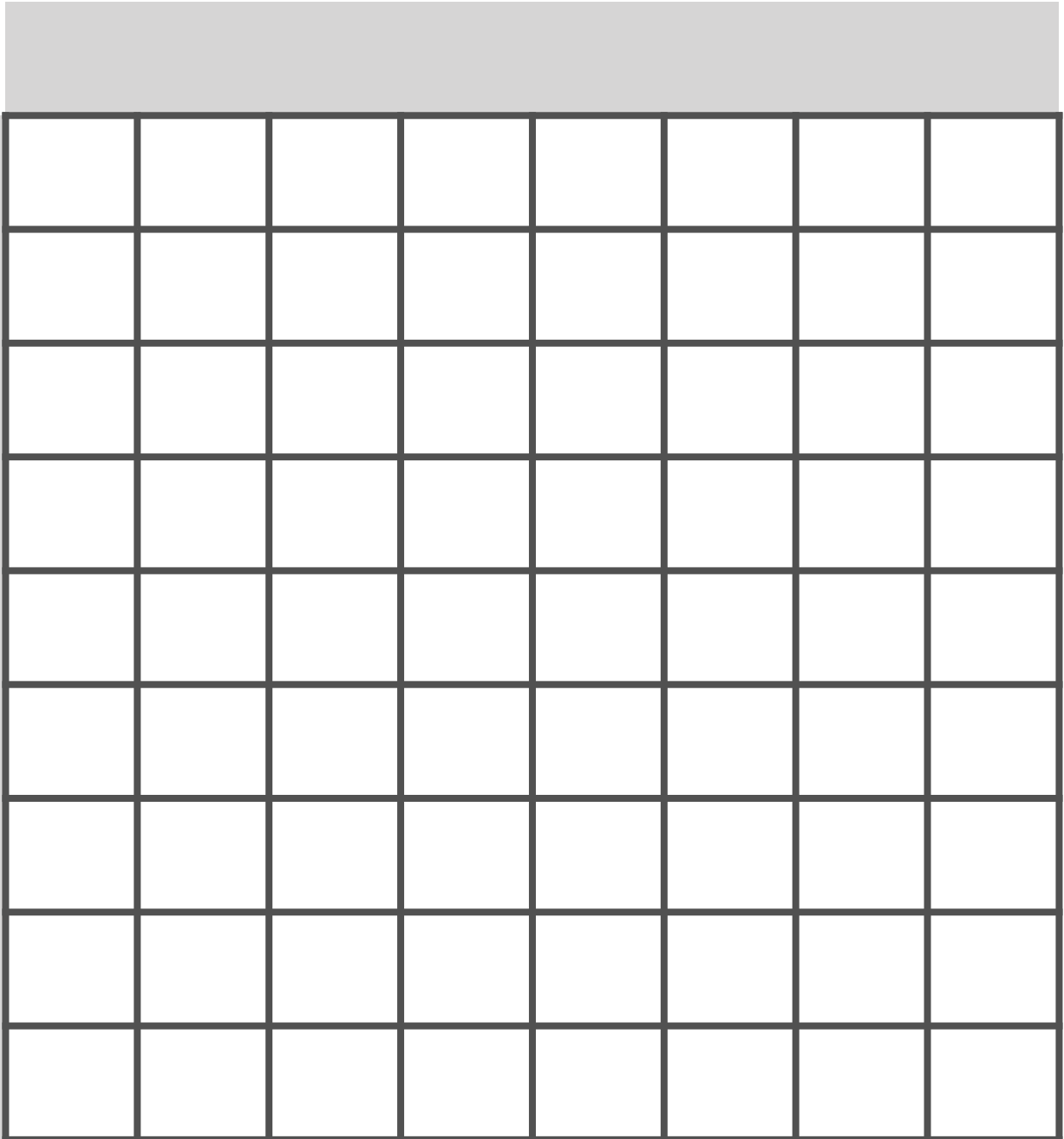
**G**


**E**


(0,0)	(1,0)	...	(n,0)
(0,1)	(1,1)	...	(n,1)
:	:		:
(0,m)	(1,m)	...	(n,m)

Find the **edit distance** between **BASEBALL** and **BALLCAP**. As a reminder the recursive formula for global alignment is given.

$$V[i, j] := \min \begin{cases} V[i-1, j-1] & \text{if } s_1[i] = s_2[j] \\ V[i-1, j-1] + 1 & \text{if } s_1[i] \neq s_2[j] \\ V[i, j-1] + 1 \\ V[i-1, j] + 1 \end{cases}$$



Edit Distance: \_\_\_\_\_

(0,0)	(1,0)	...	(n,0)
(0,1)	(1,1)	...	(n,1)
:	:		:
(0,m)	(1,m)	...	(n,m)

Build the **Suffix Tree** for the string `MISSISSIPPI` using Ukkonen's algorithm. Without edge compression or suffix links. When you have completed the implicit tree for a given value, circle it in the table below.

											1	1	1
1	2	3	4	5	6	7	8	9	0	1	2		
<b>M</b>	<b>I</b>	<b>S</b>	<b>S</b>	<b>I</b>	<b>S</b>	<b>S</b>	<b>I</b>	<b>P</b>	<b>P</b>	<b>I</b>	<b>\$</b>		