Nucleotide Sequence Alignment Part 2

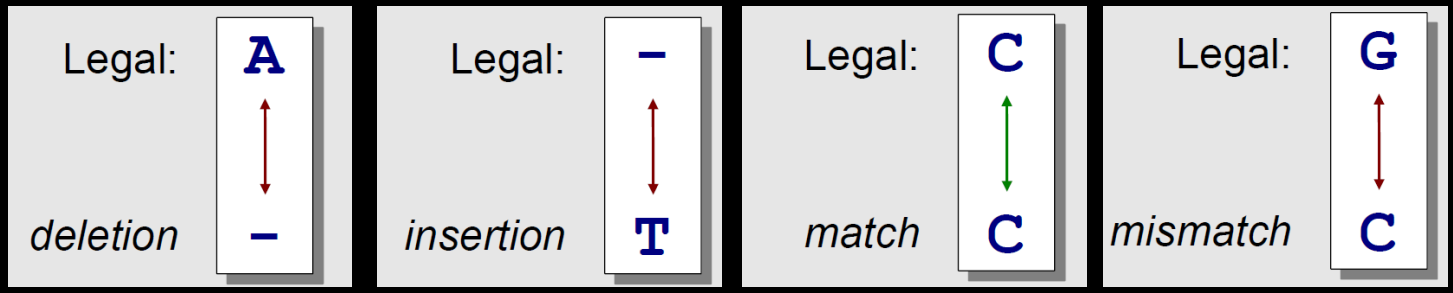
Some slides contributed by Daniel Lopresti

Major stages of sequence alignment

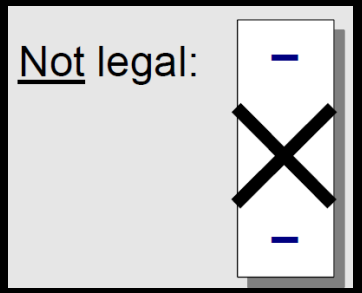
- Parsing (read the data)
- Scoring
  - Fill the values of the m x n grid based on alignment score
    - Looking up the score on the substitution matrix
    - Different kinds of substitution matrices
    - Gap penalties
- Backtracking
  - Following the chain of maximum scores backwards to determine the actual alignment of nucleotides.
- Output (print the result)

On scoring

- Not every kind of alignment is allowed
  - These are okay – they have biological explanations



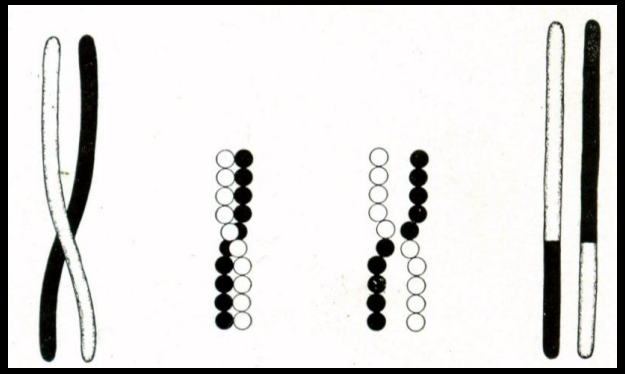
  - Gap to gap is <u>not okay</u>
    - Exists in theory, but has no biological relevance

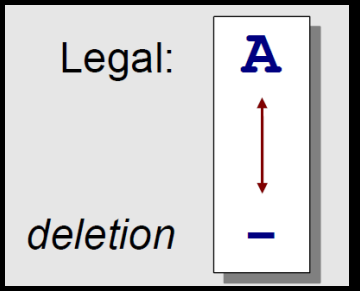


  - This also makes no sense in the context of the grid

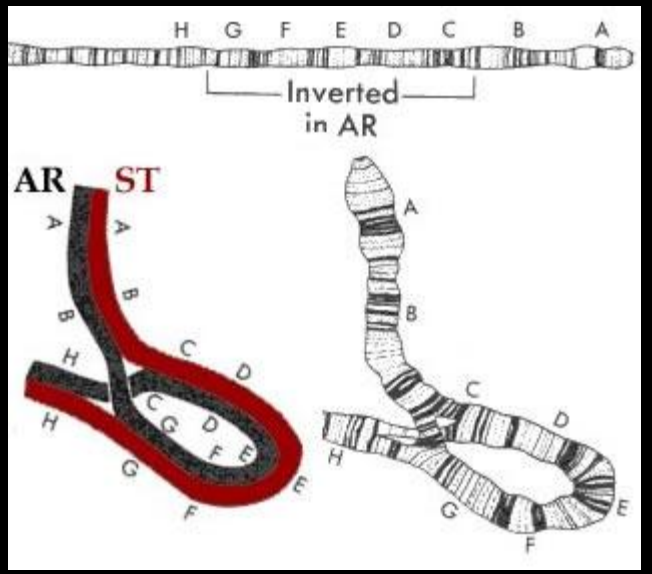
How deletions happen

- Chromosomal crossovers
  - Matching regions of a homologous chromosome break and rejoin during gamete cell division
  - Unequal crossovers result in deletions

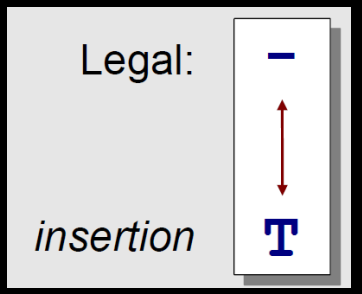




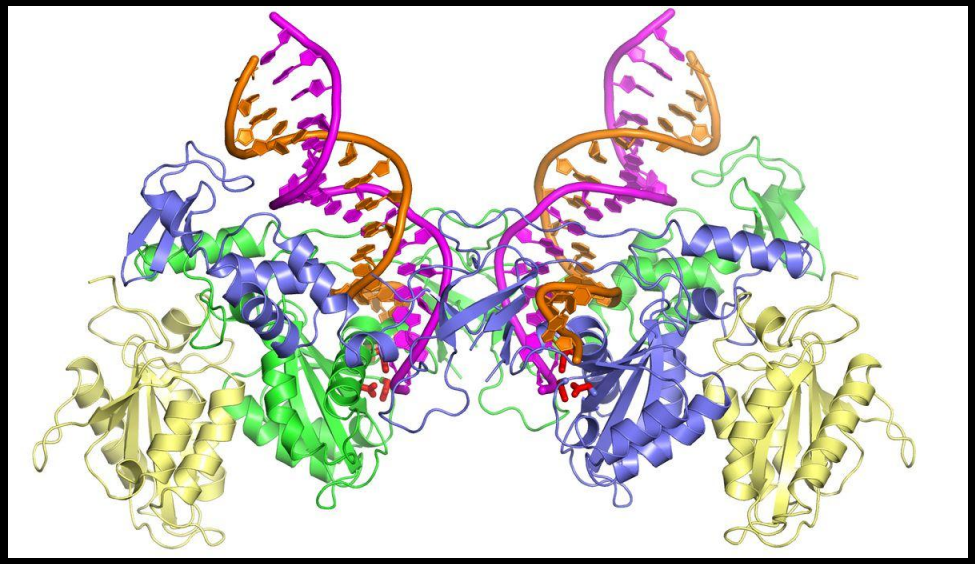
- Crossovers in an inversion can frequently lead to deletions, because of curvature



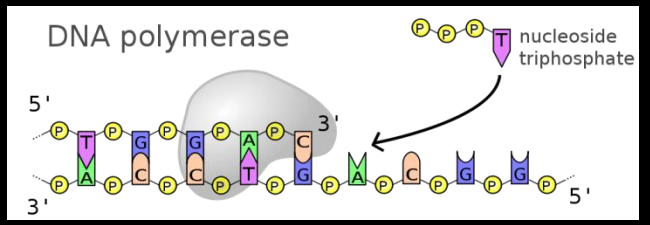
How Insertions can happen



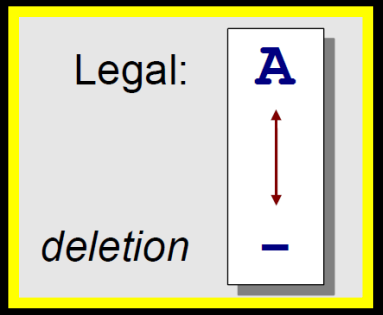
- Retroviral activity
  - Retroviral Integrases (like HIV integrase, shown below) help insert Viral DNA into host DNA, so that the host creates viral proteins

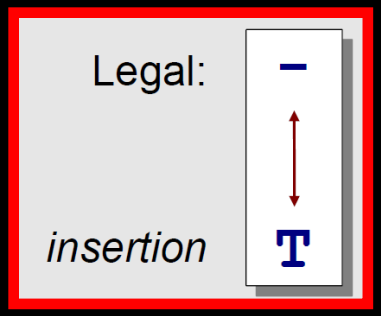


- Slippage in DNA polymerase
  - In replication, polymerase can accidentally add extra base(s)

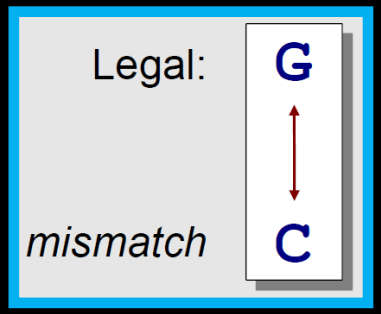


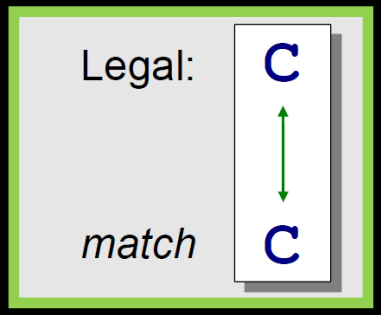
The basis for scoring





	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

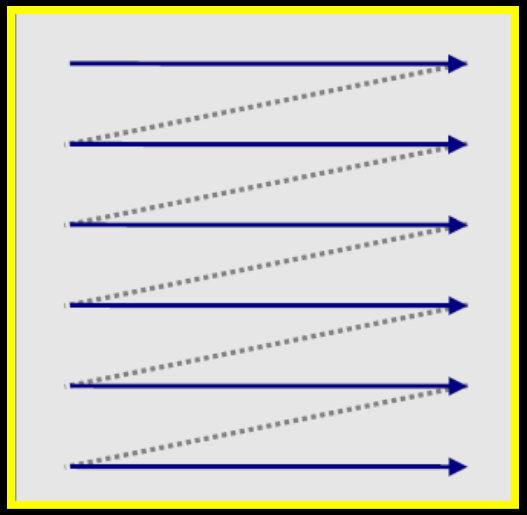


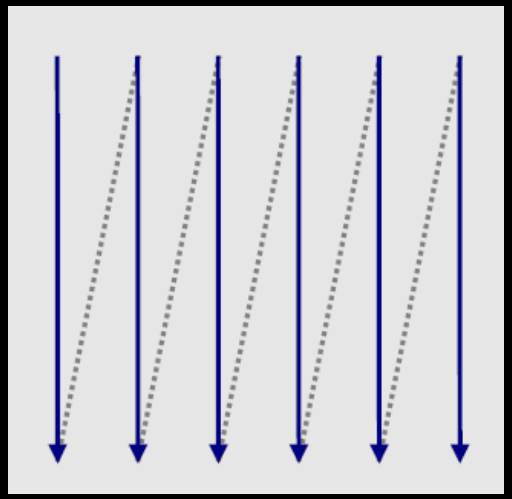


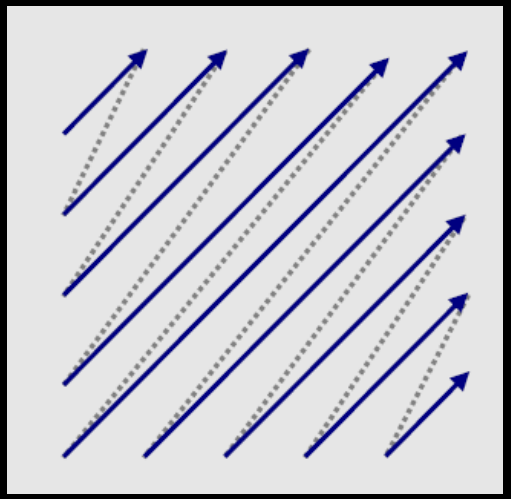
Scoring without gap penalties

		G	G	A	T	C
	0	.	.	.	.	.
C	.	.	.	.	.	.
G	.	.	.	.	.	.
C	.	.	.	.	.	.

Note: there are several orders to score in

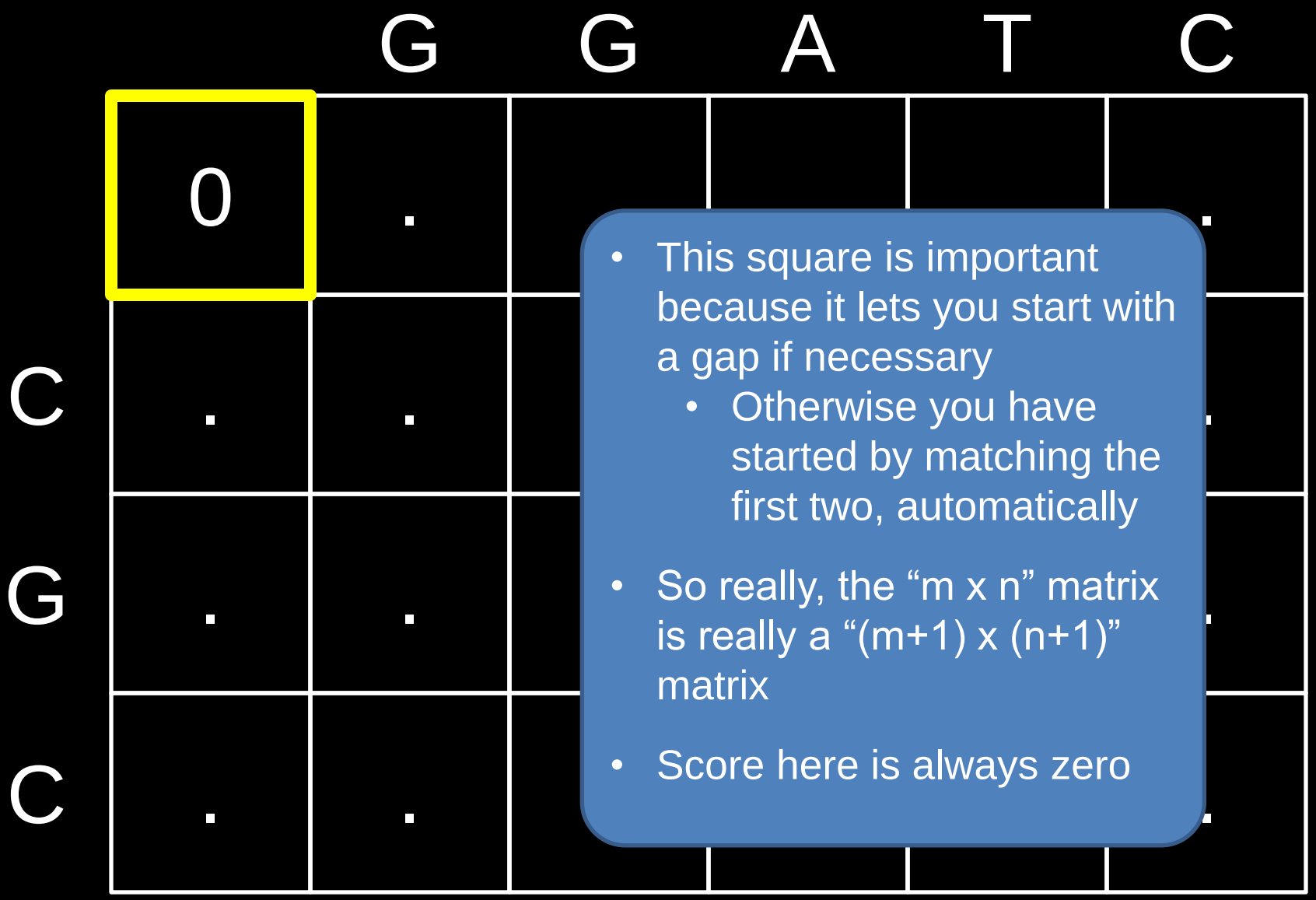






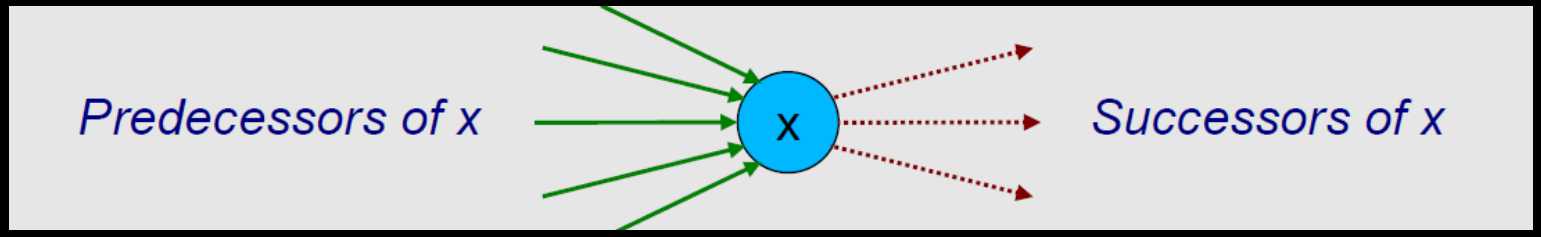
- All of these orders are the same
- Scores depend on scores to the left, up, and upper left
- Be careful how you do this to avoid confusion (e.g. bugs)
- This is how I am going to run the example

The (very) important first square



- This square is important because it lets you start with a gap if necessary
  - Otherwise you have started by matching the first two, automatically
- So really, the “m x n” matrix is really a “(m+1) x (n+1)” matrix
- Score here is always zero

An important generalization of the method

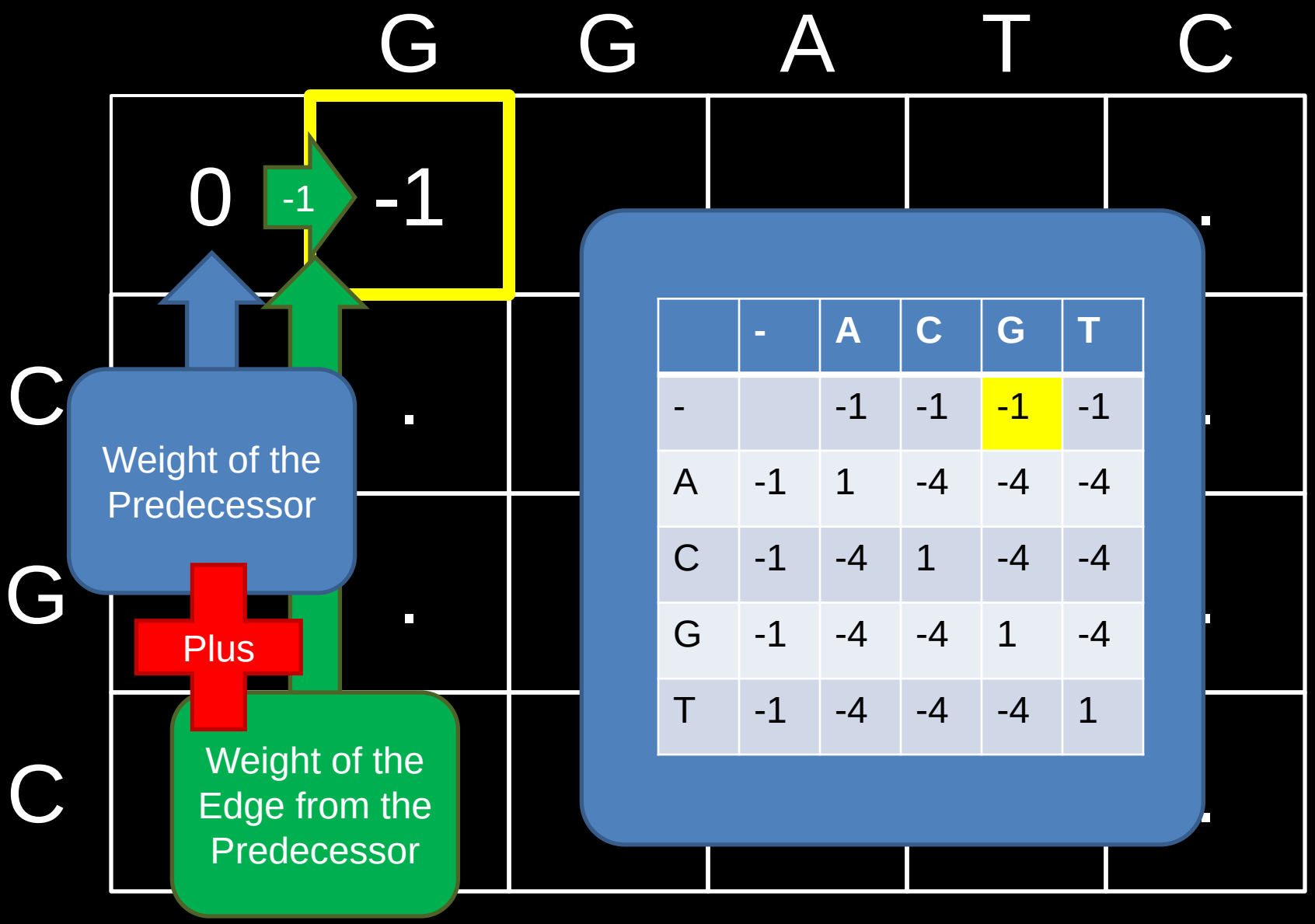


- You can think of each box on the grid as a node, connected to predecessors and successors.

$$s_x = \max \text{ of } \begin{cases} s_y + \text{weight of edge } (y, x) \text{ where} \\ \qquad y \in \text{Predecessors}(x) \end{cases}$$

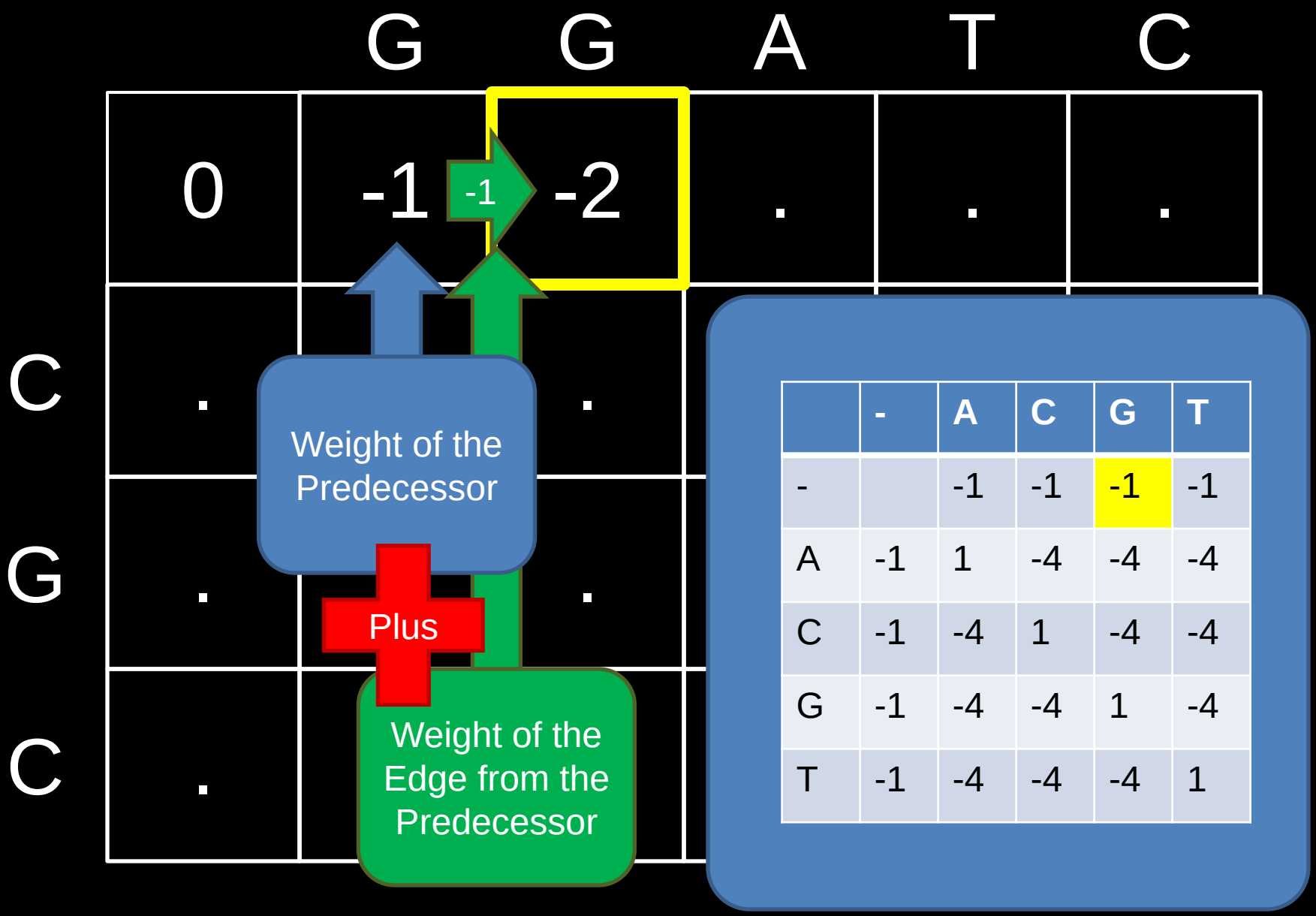
- It is very easy to write bugs where you add values from the wrong predecessors or edges.
- Not every node has the same # or “kind” of edge

Step 2: Guanine is aligned with a gap



	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 3: A second G is aligned with a gap



	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 4: Adenine is aligned with a gap

		G	G	A	T	C
	0	-1	-2	-3	.	.
C	.	.	.			
G	.	.	.			
C	.	.	.			

-1

	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 5: Thymine is aligned with a gap

		G	G	A	T	C
	0	-1	-2	-3	-4	.
C	.	.	.			
G	.	.	.			
C	.	.	.			

-1

	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

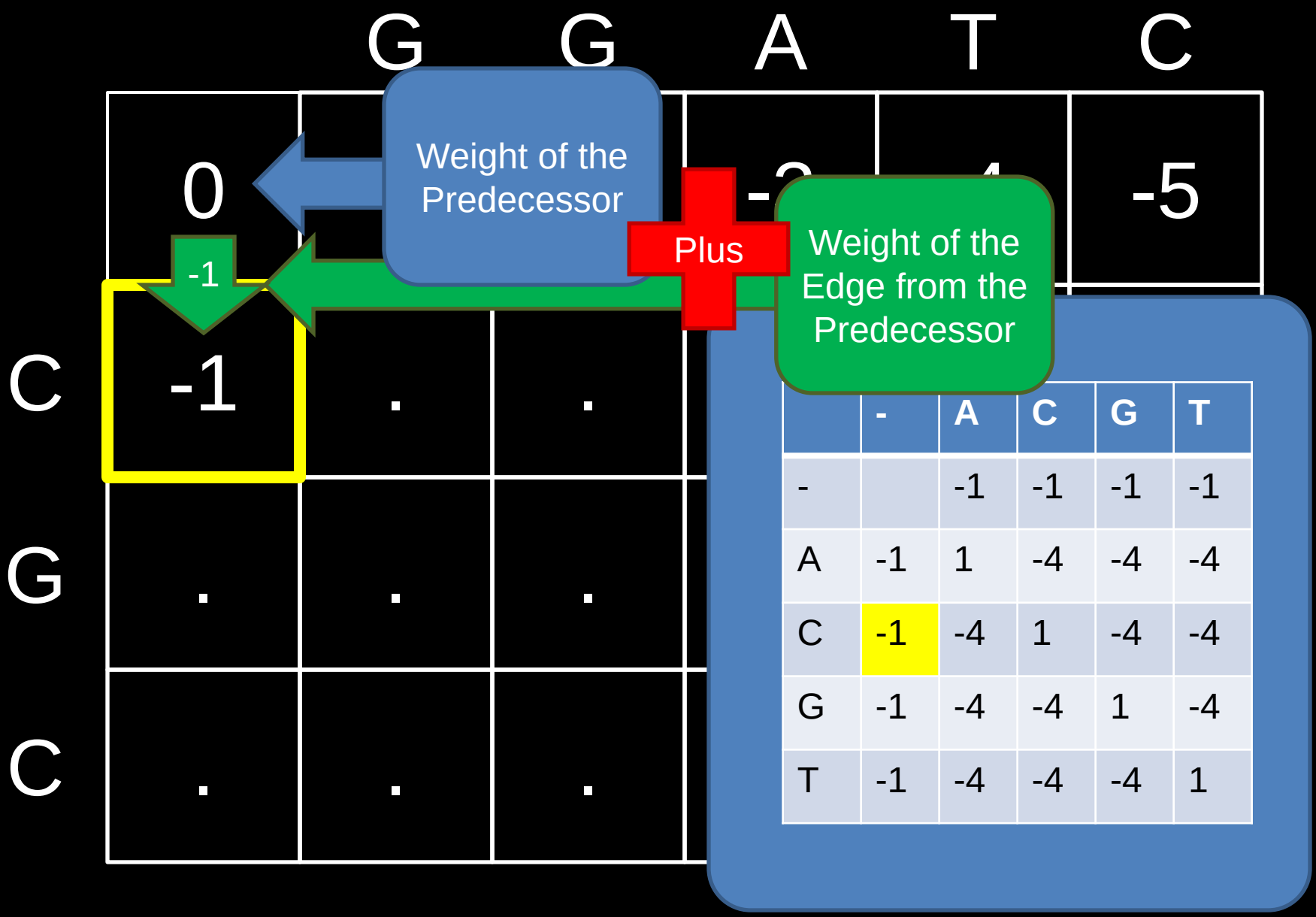
Step 6: Cytosine is aligned with a gap

		G	G	A	T	C
	0	-1	-2	-3	-4	-5
C	.	.	.			
G	.	.	.			
C	.	.	.			

-1

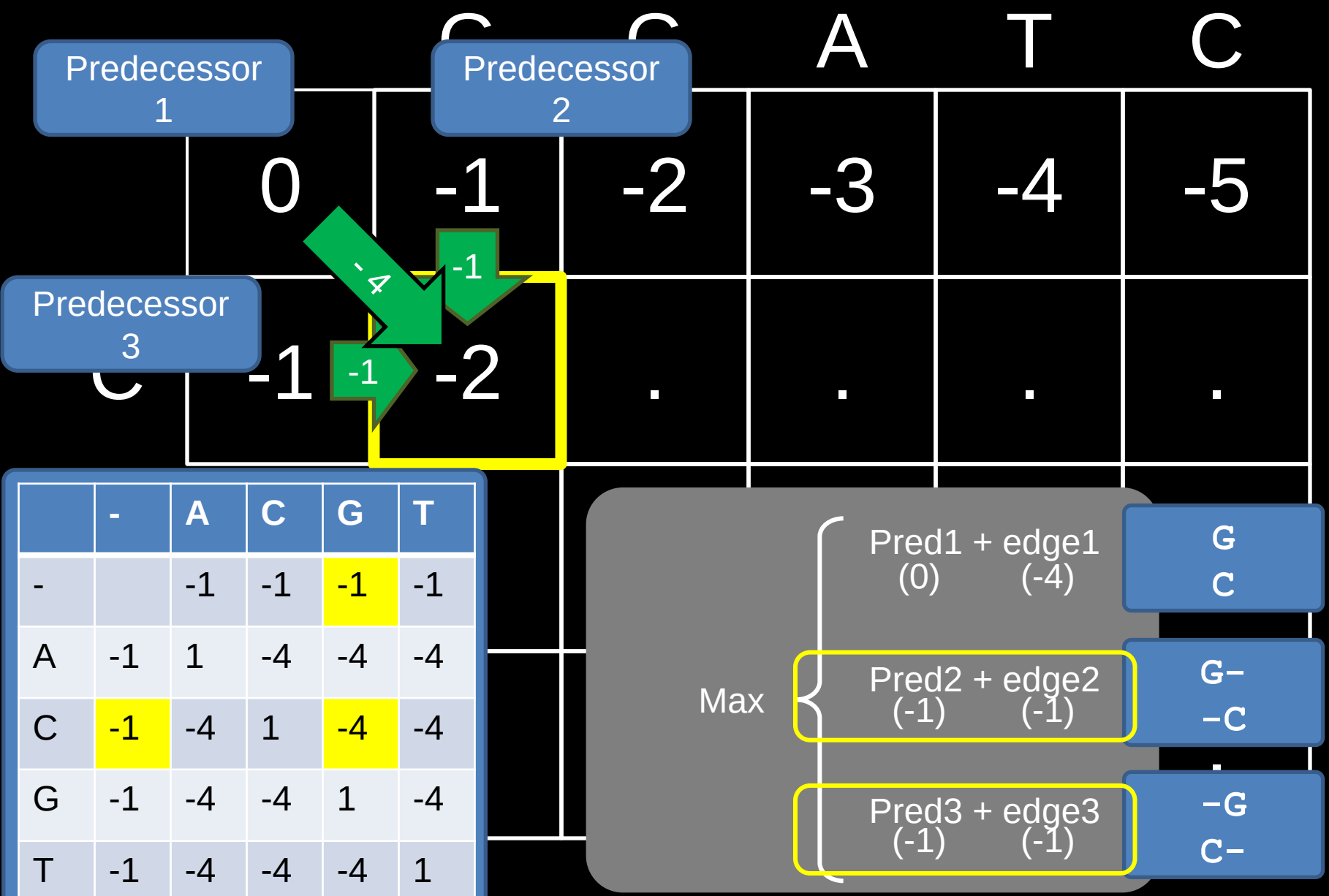
	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 7: Cytosine is aligned with a gap



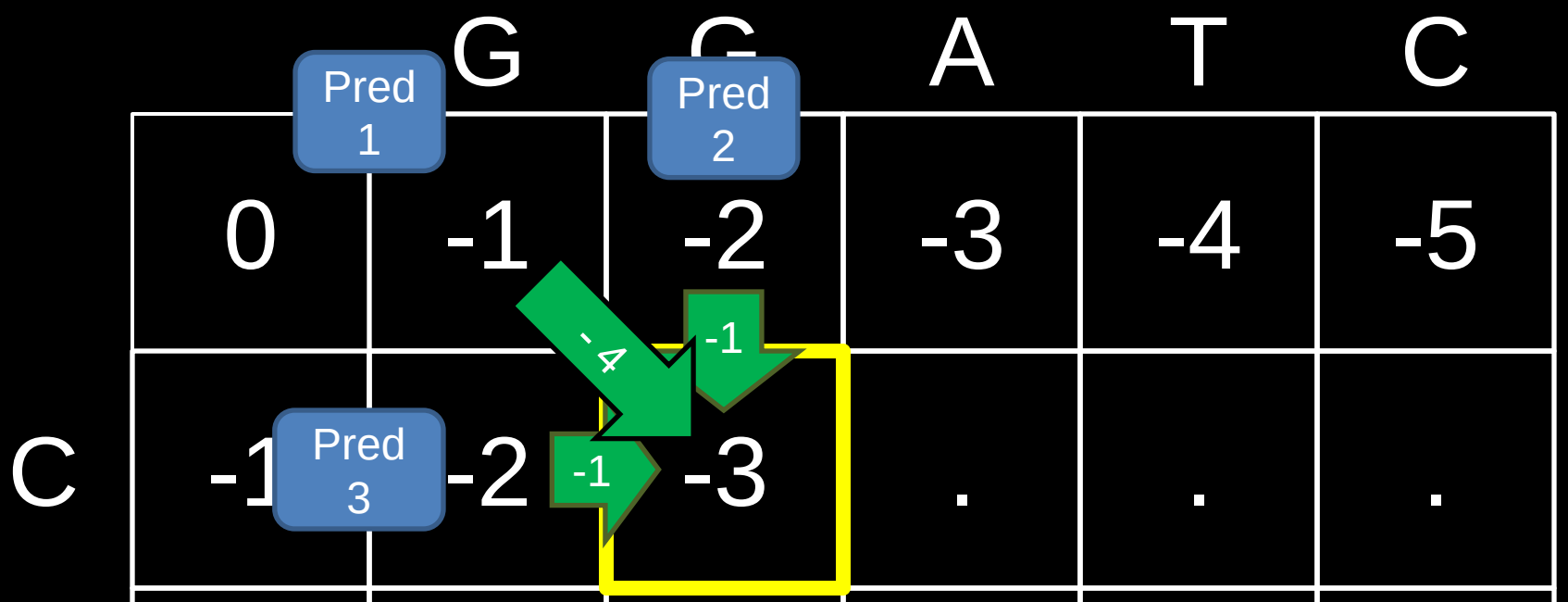
	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 8: Take the max of 3 possibilities



	-	A	C	G	T
-		-1	-1	-1	-1
A	-1	1	-4	-4	-4
C	-1	-4	1	-4	-4
G	-1	-4	-4	1	-4
T	-1	-4	-4	-4	1

Step 9: Take the max of 3 possibilities



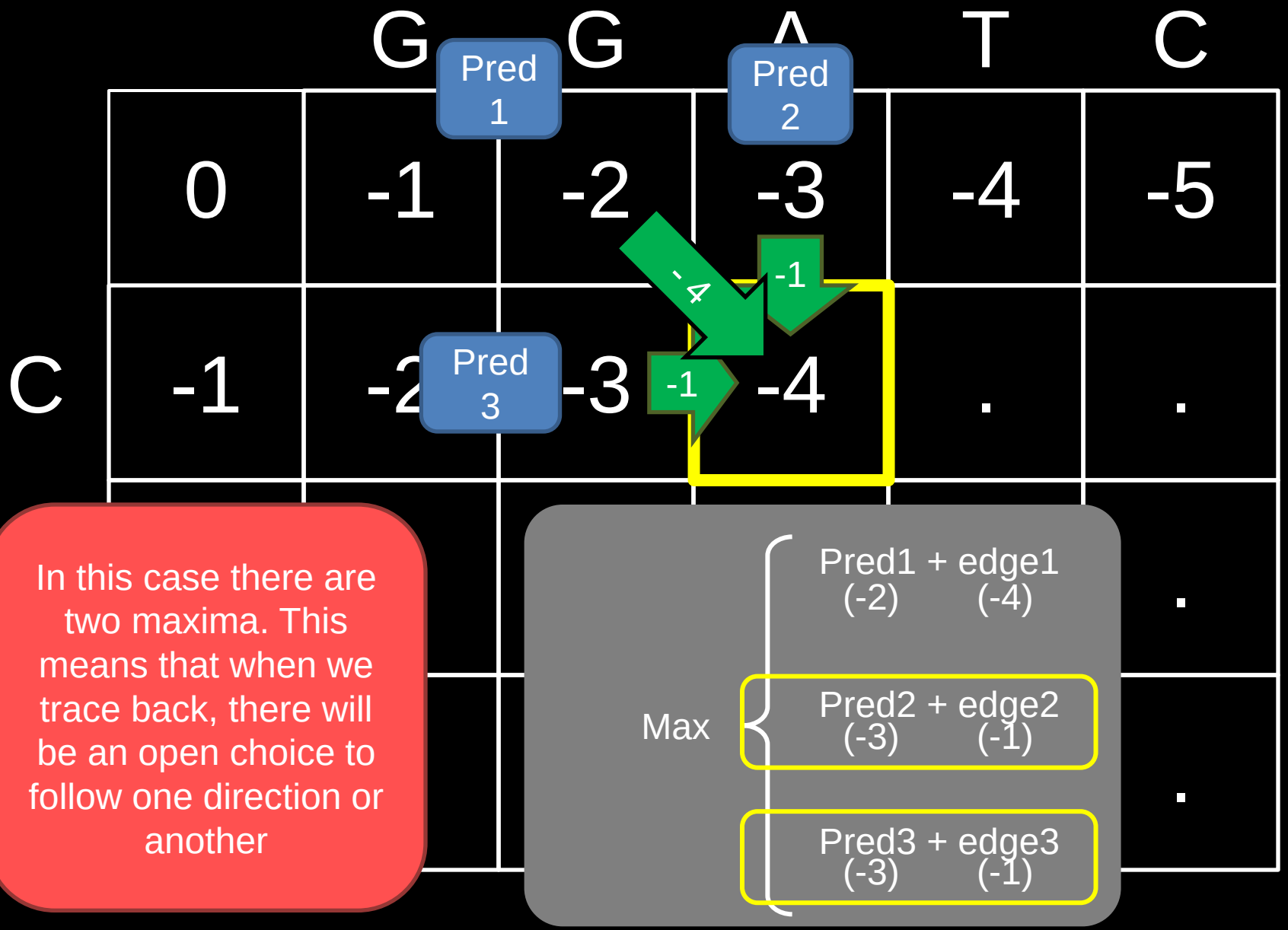
The easy thing to get confused about at this step is that there might be multiple ways to get to the (1,1) square.

You just care about the value it has: -2.

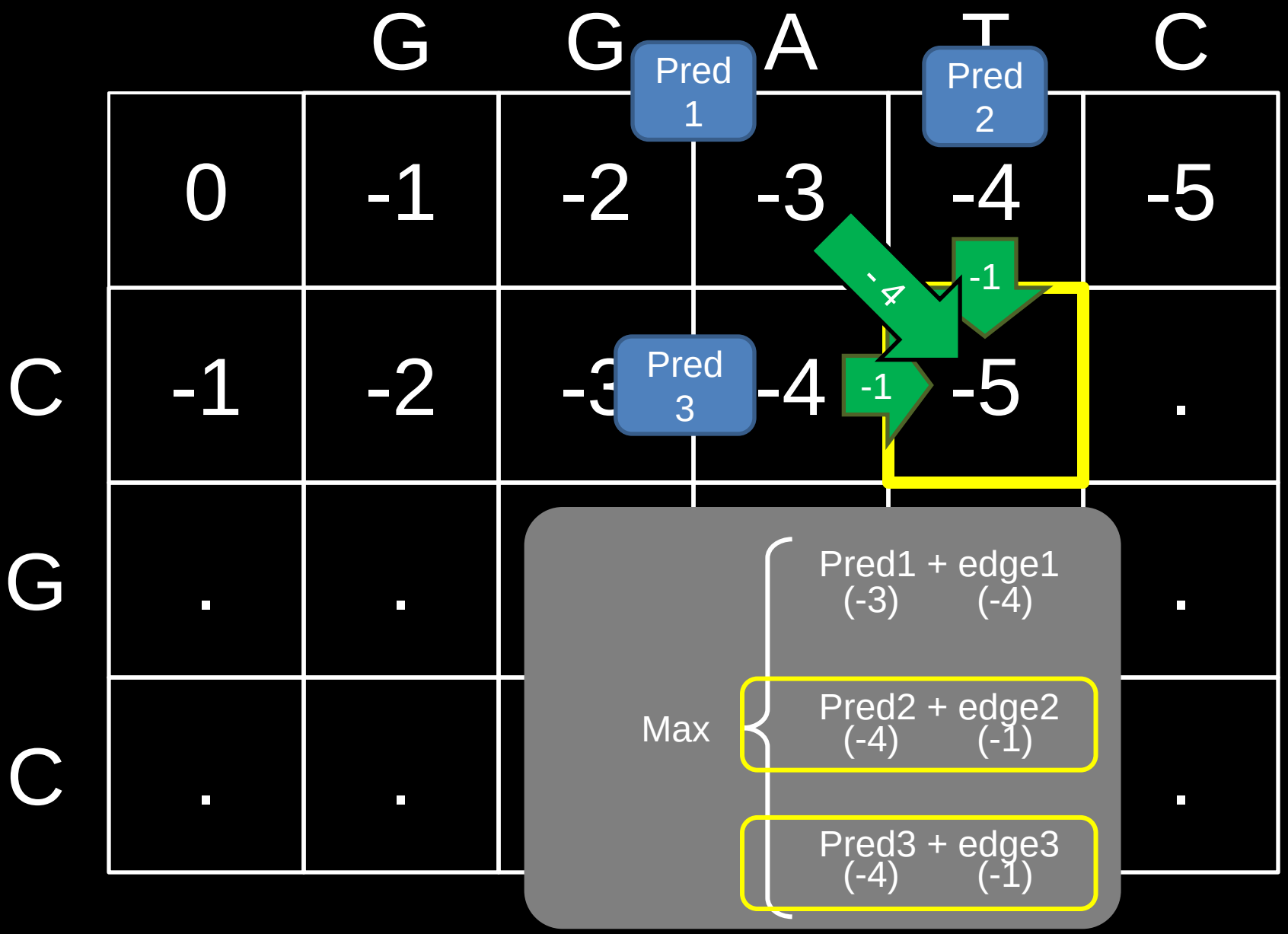
Max

Pred1 + edge1	(-1) (-4)	GG –C
Pred2 + edge2	(-2) (-1)	GG– ––C
Pred3 + edge3	(-2) (-1)	–GG C––
		G–G –C–

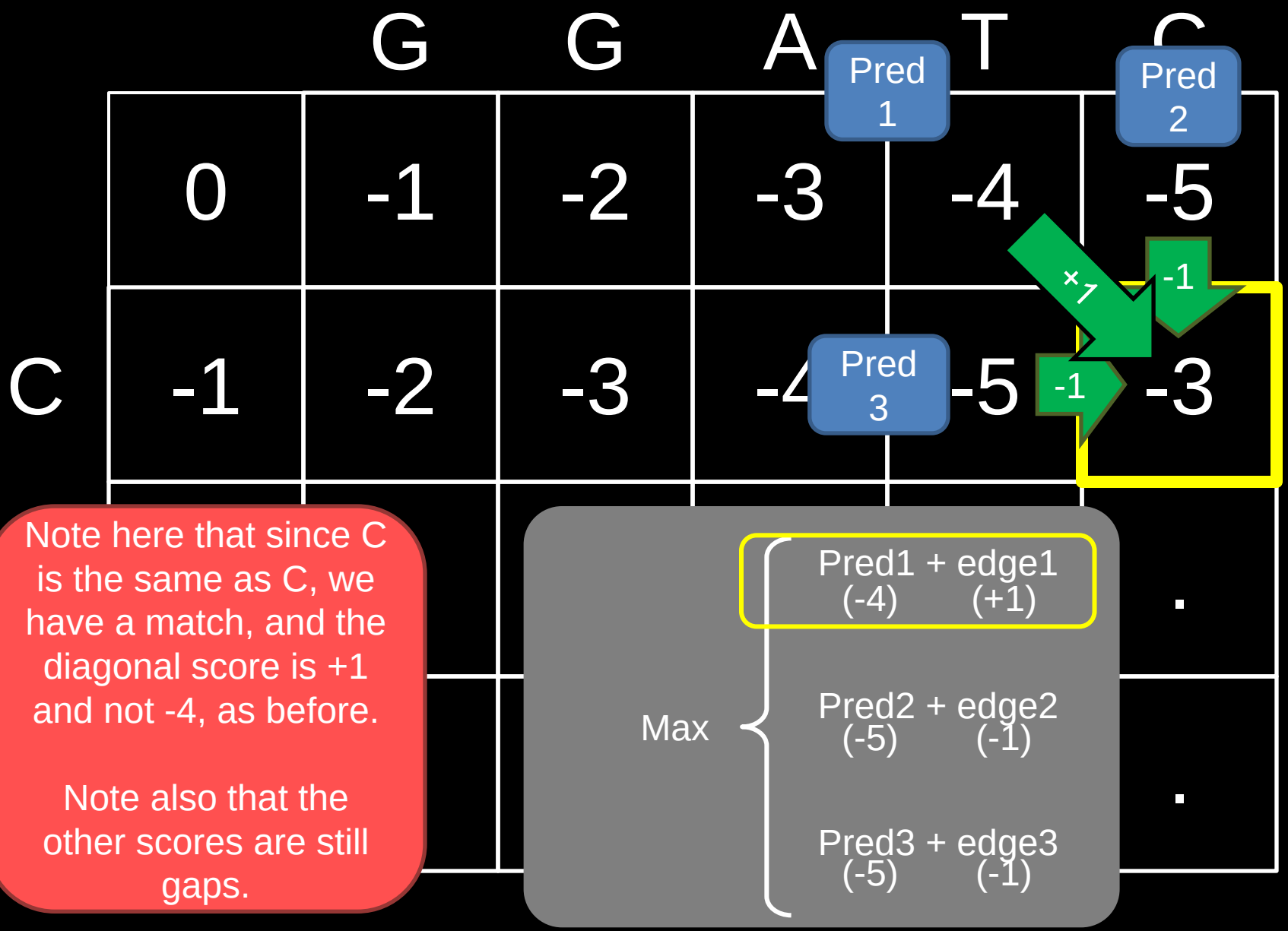
Step 9: Take the max of 3 possibilities



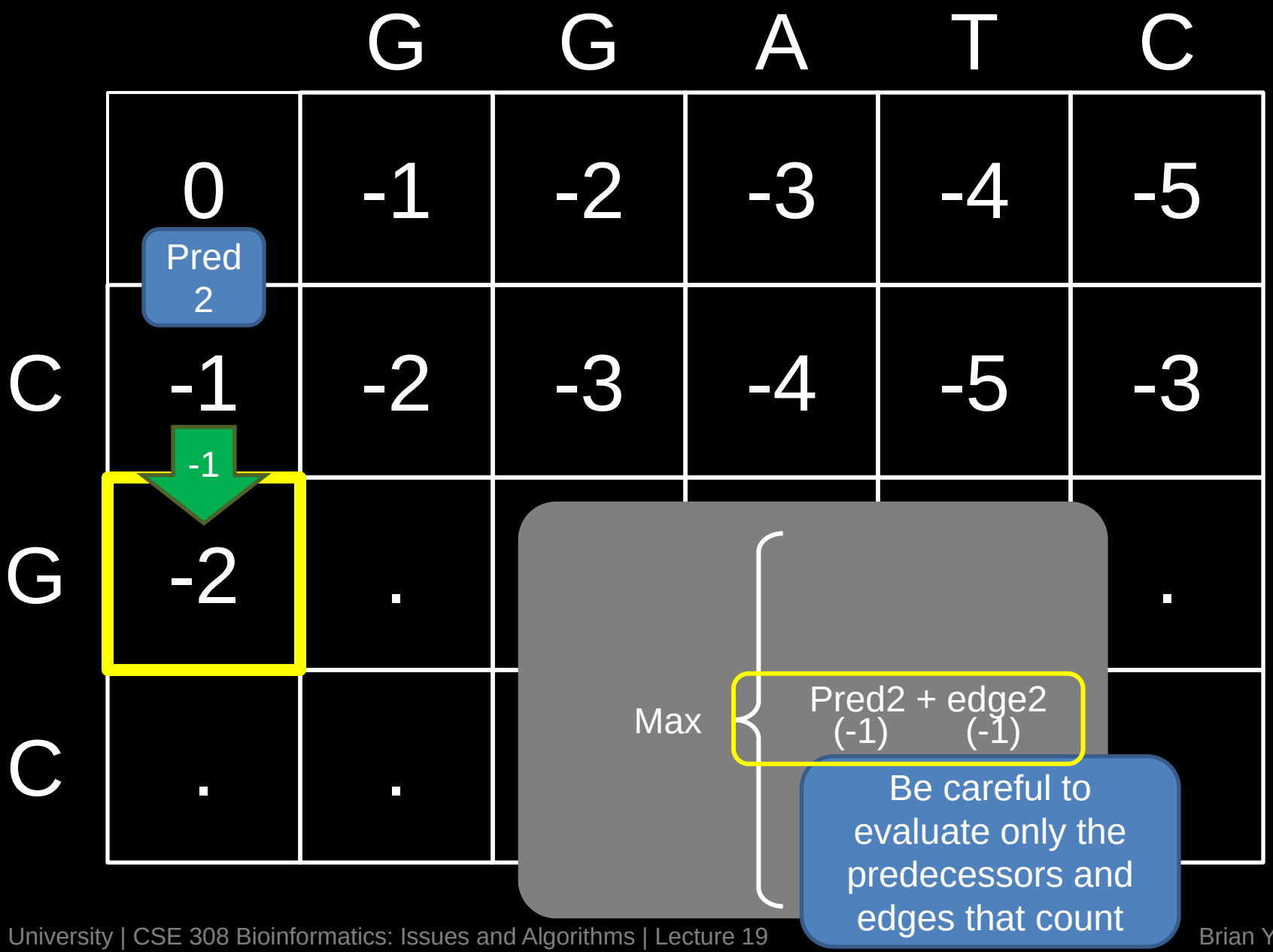
Step 10: Take the max of 3 possibilities



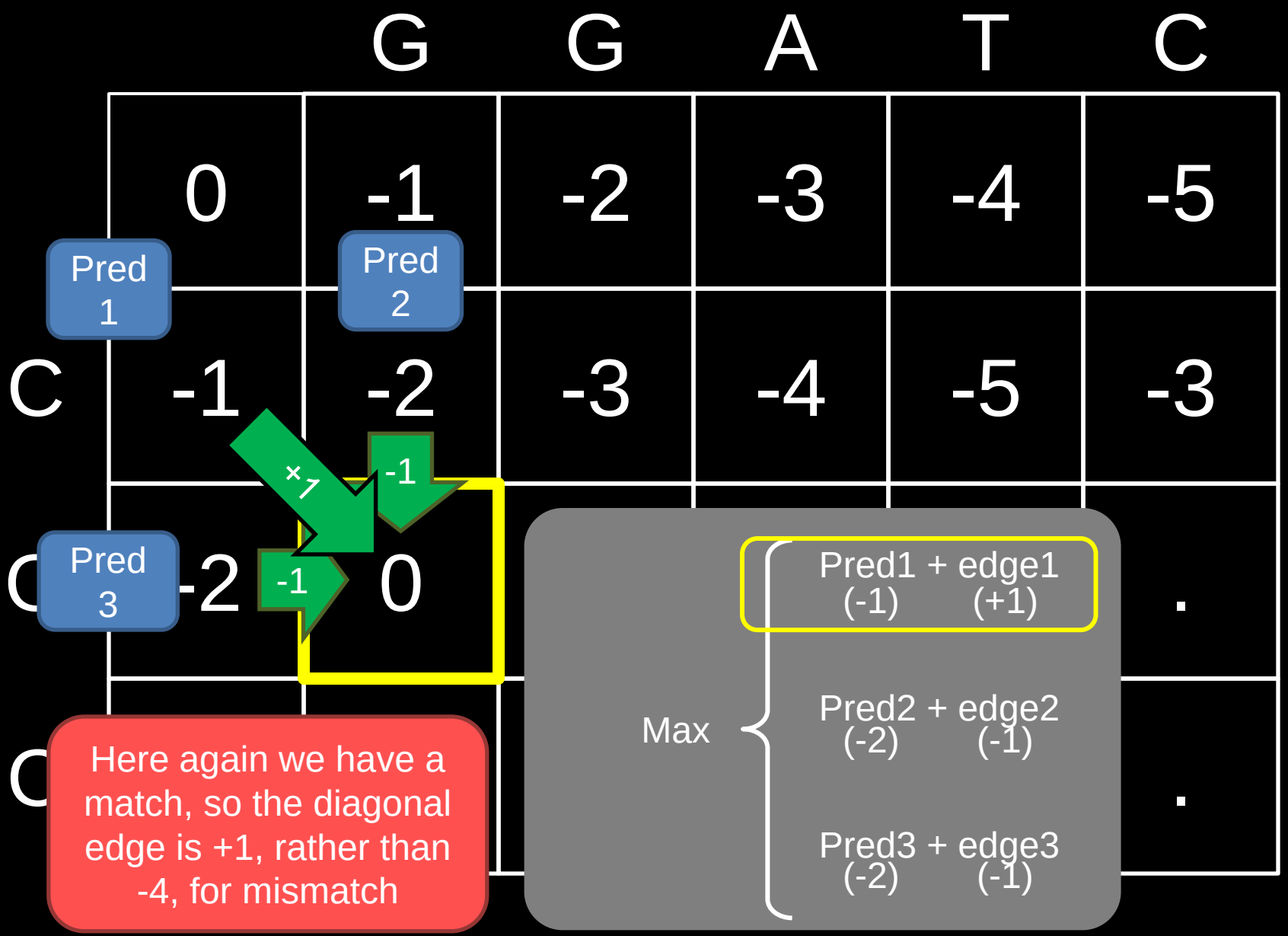
Step 11: Take the max of 3 possibilities



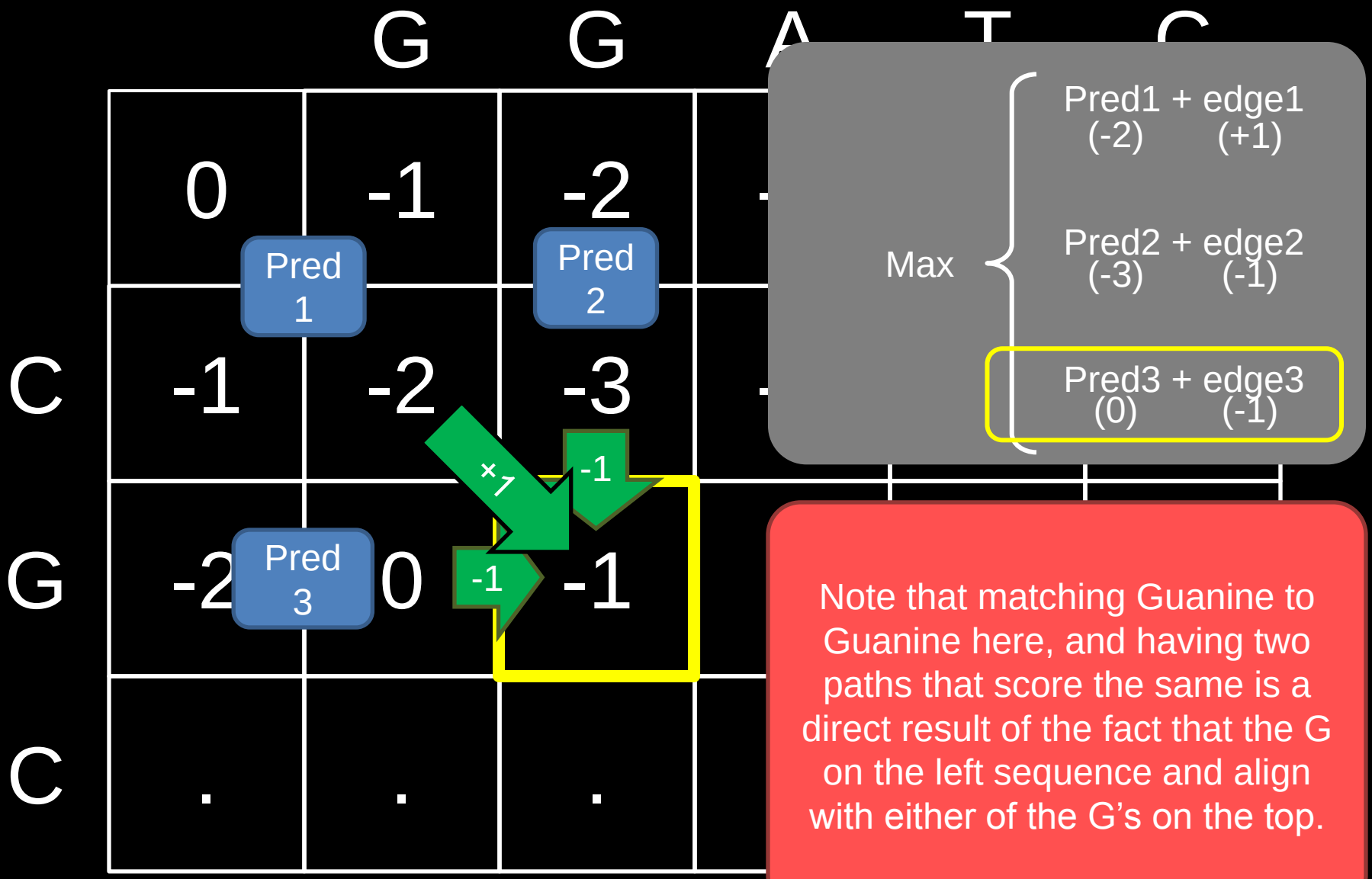
Step 12: Take the max of 3 possibilities



Step 13: Take the max of 3 possibilities

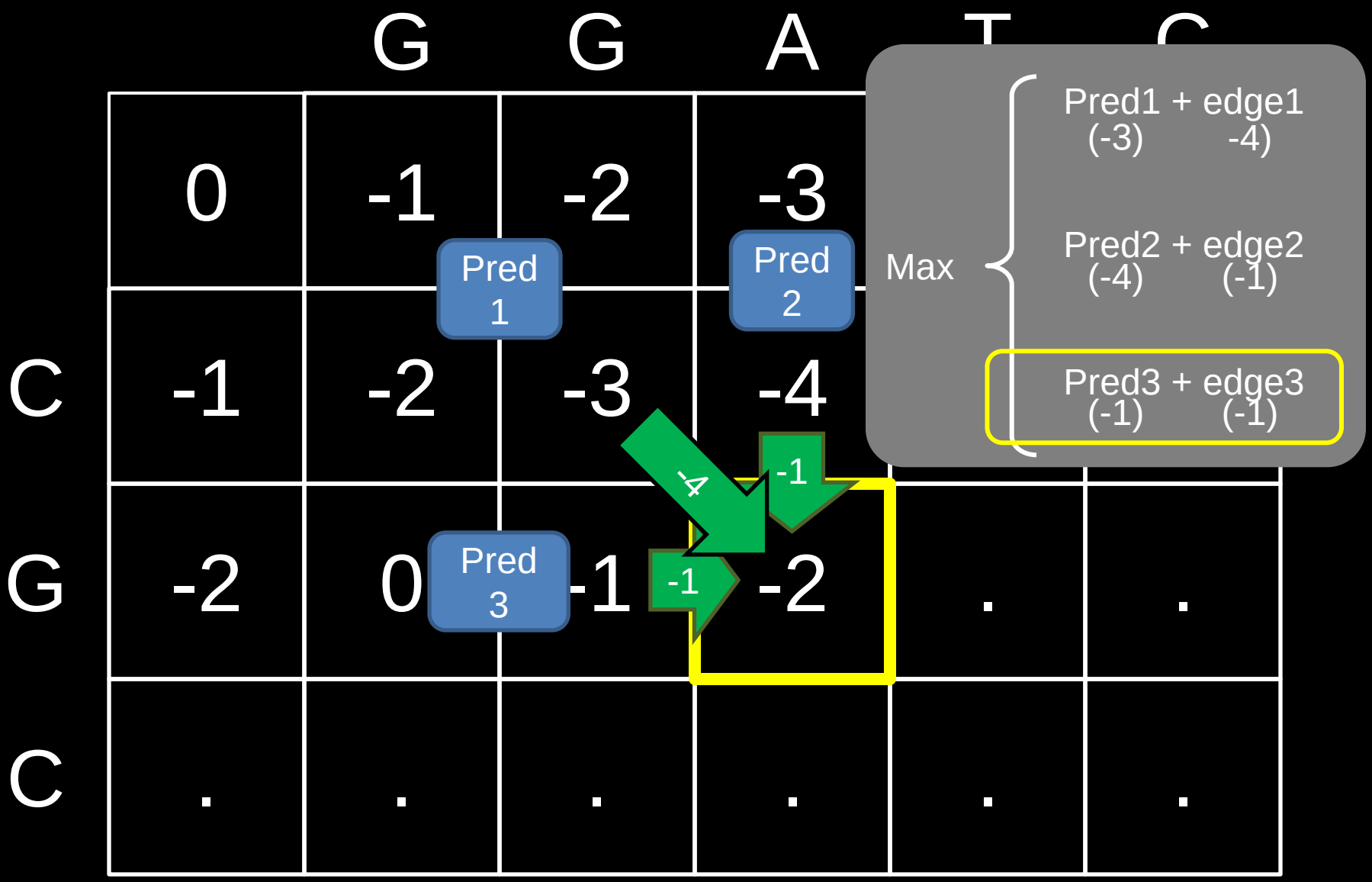


Step 14: Take the max of 3 possibilities



Note that matching Guanine to Guanine here, and having two paths that score the same is a direct result of the fact that the G on the left sequence and align with either of the G's on the top.

Step 15: Take the max of 3 possibilities



Step 16: Finishing off the grid

		G	G	A	T	C
	0	-1	-2	-3	-4	-5
C	-1	-2	-3	-4	-5	-3
G	-2	0	-1	-2	-3	-4
C	-3	-1	-2	-3	-4	-2

What exactly we've done

- We have scored the grid based on substitution rules, and a linear gap penalty

- Later in this lecture we will see how to score based on an affine gap penalty

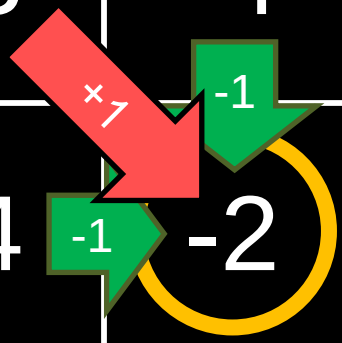
Backtracking Step 1: Begin with highest val

		G	G	A	T	C
	0	-1	-2	-3	-4	-5
C	-1	-2	-3	-4	-5	-3
G	-2	0	-1	-2	-3	-4
C	-3	-1	-2	-3	-4	-2

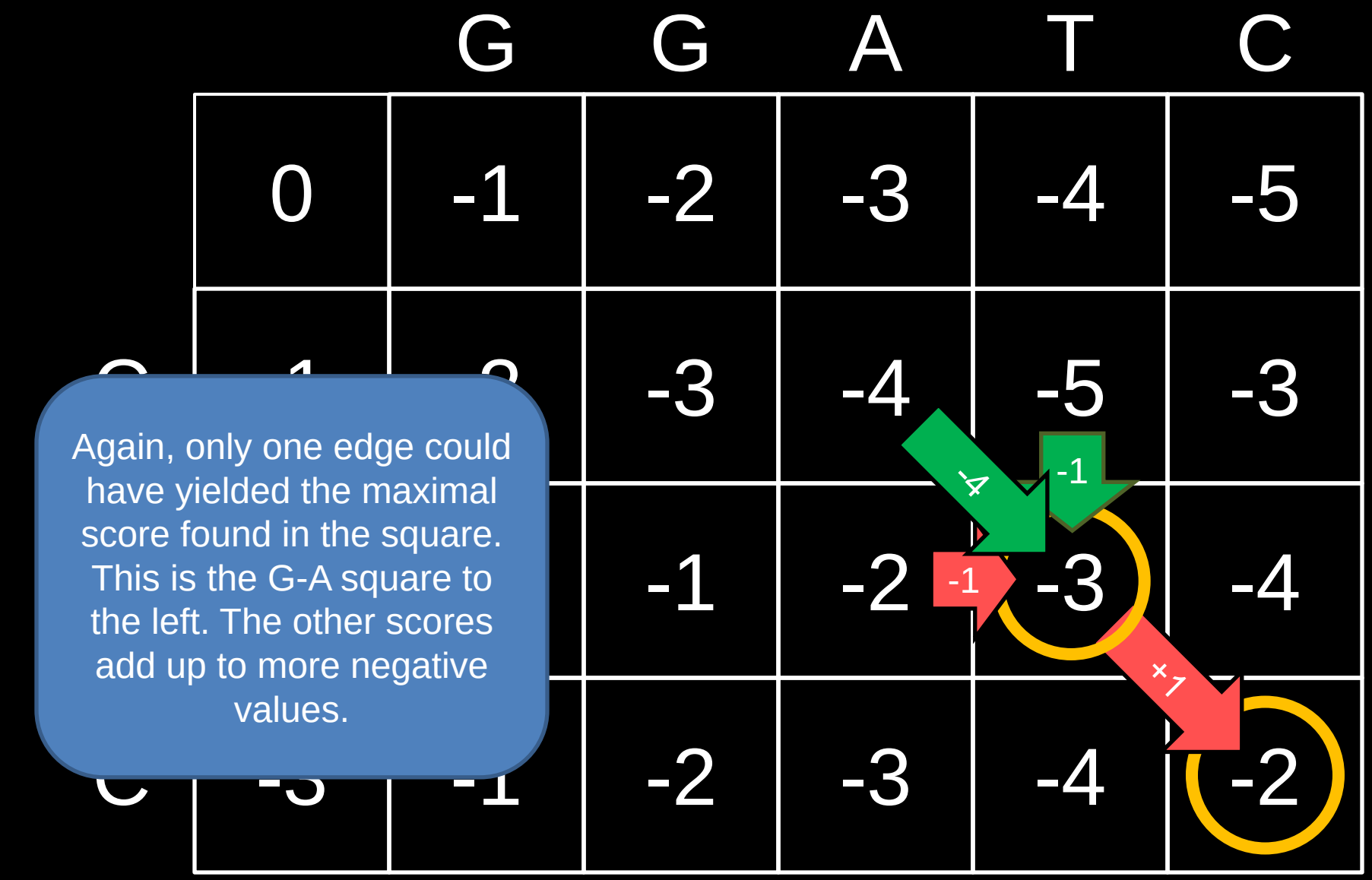
Backtracking Step 2: Find the path we took

		G	G	A	T	C
	0	-1	-2	-3	-4	-5
G	-1	-2	-3	-4	-5	-3
G	-2	-1	-1	-2	-3	-4
C	-3	-1	-2	-3	-4	-2

You can backtrack from G(i,j) to G(i-1, j), G(i, j-1), and G(i-1, j-1). In this case, only one of these squares yields an edge that is maximal.

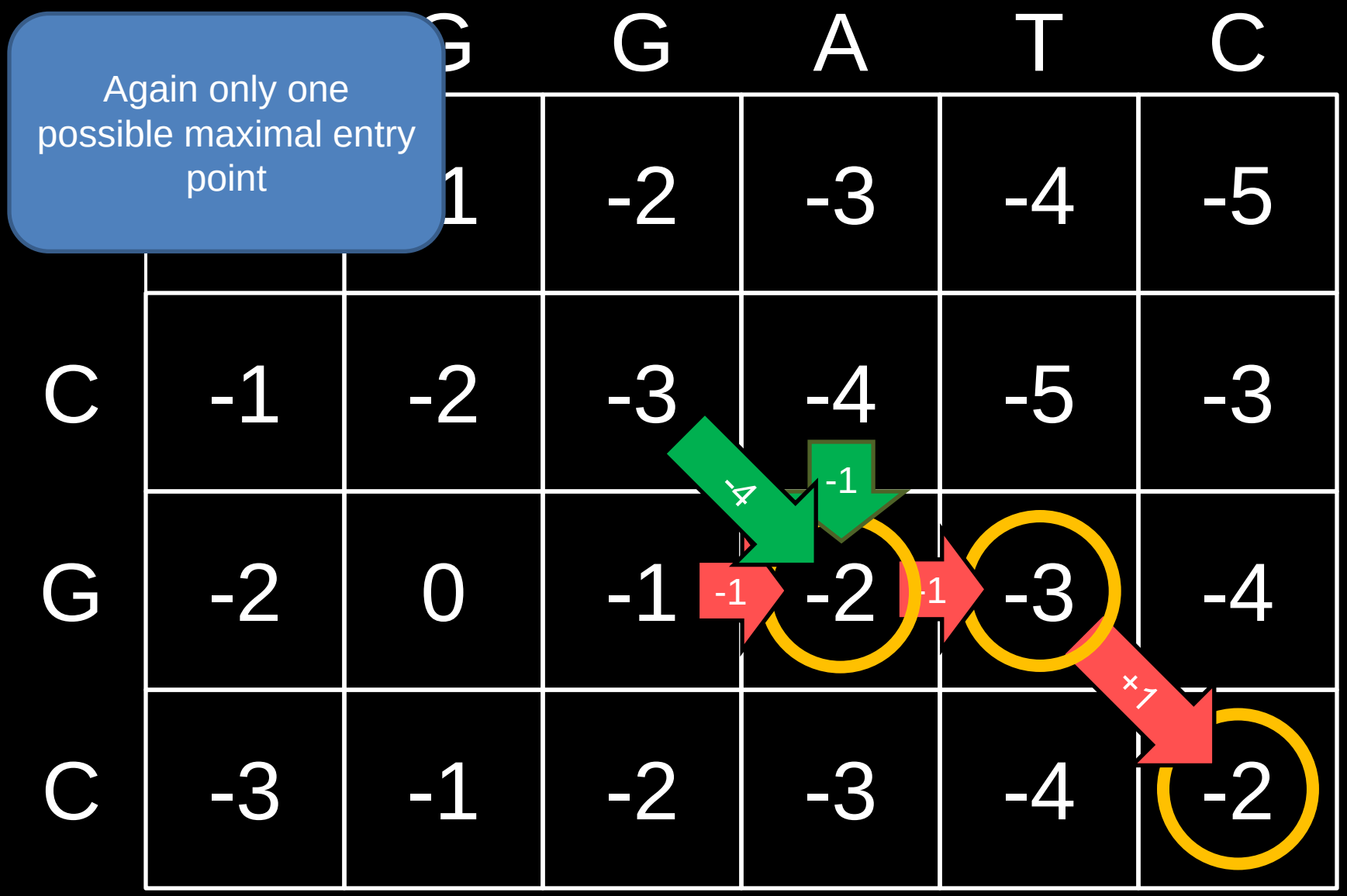


Backtracking Step 2: Find the path we took

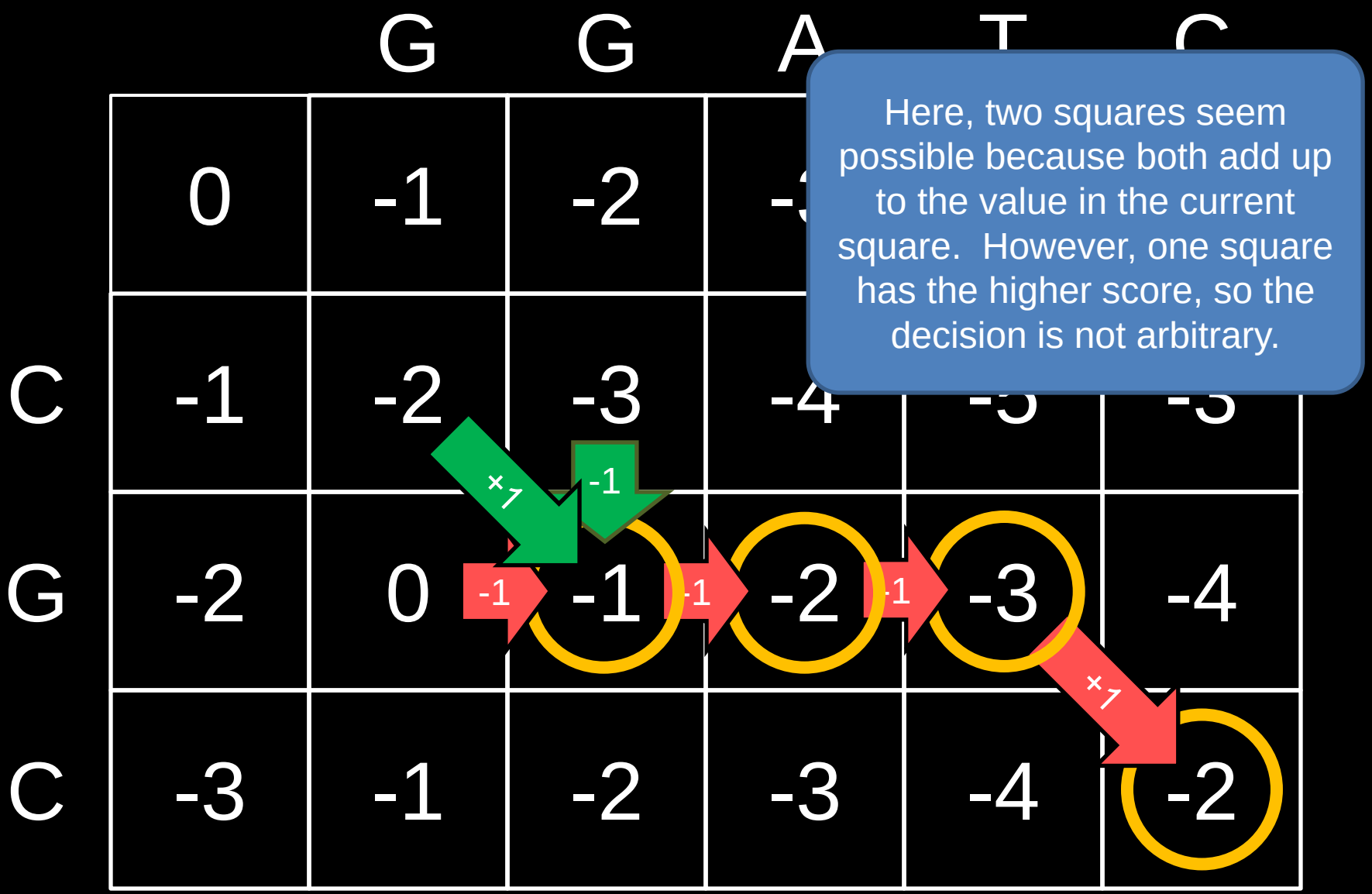


Backtracking Step 3: Find the path we took

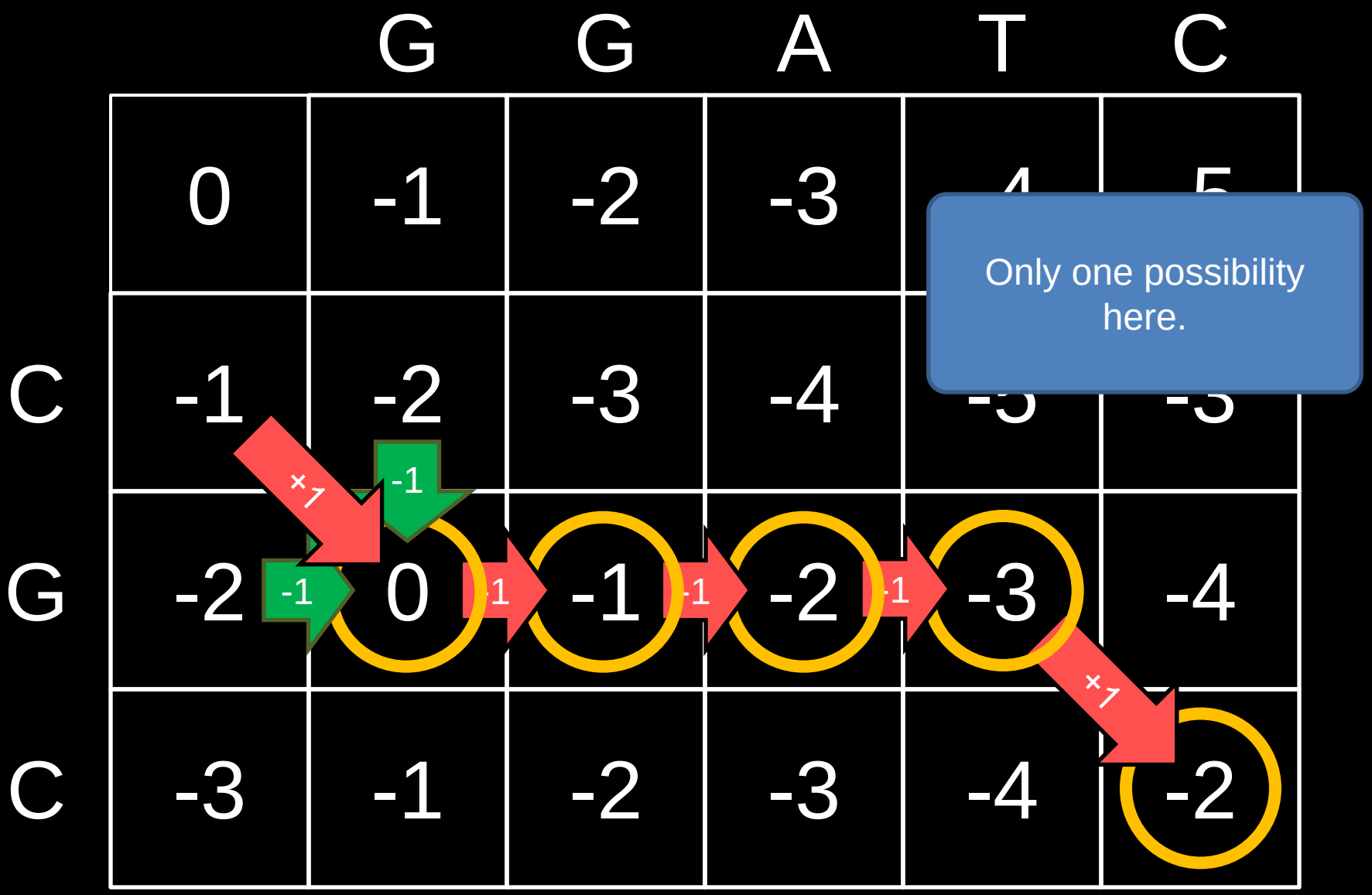
Again only one possible maximal entry point



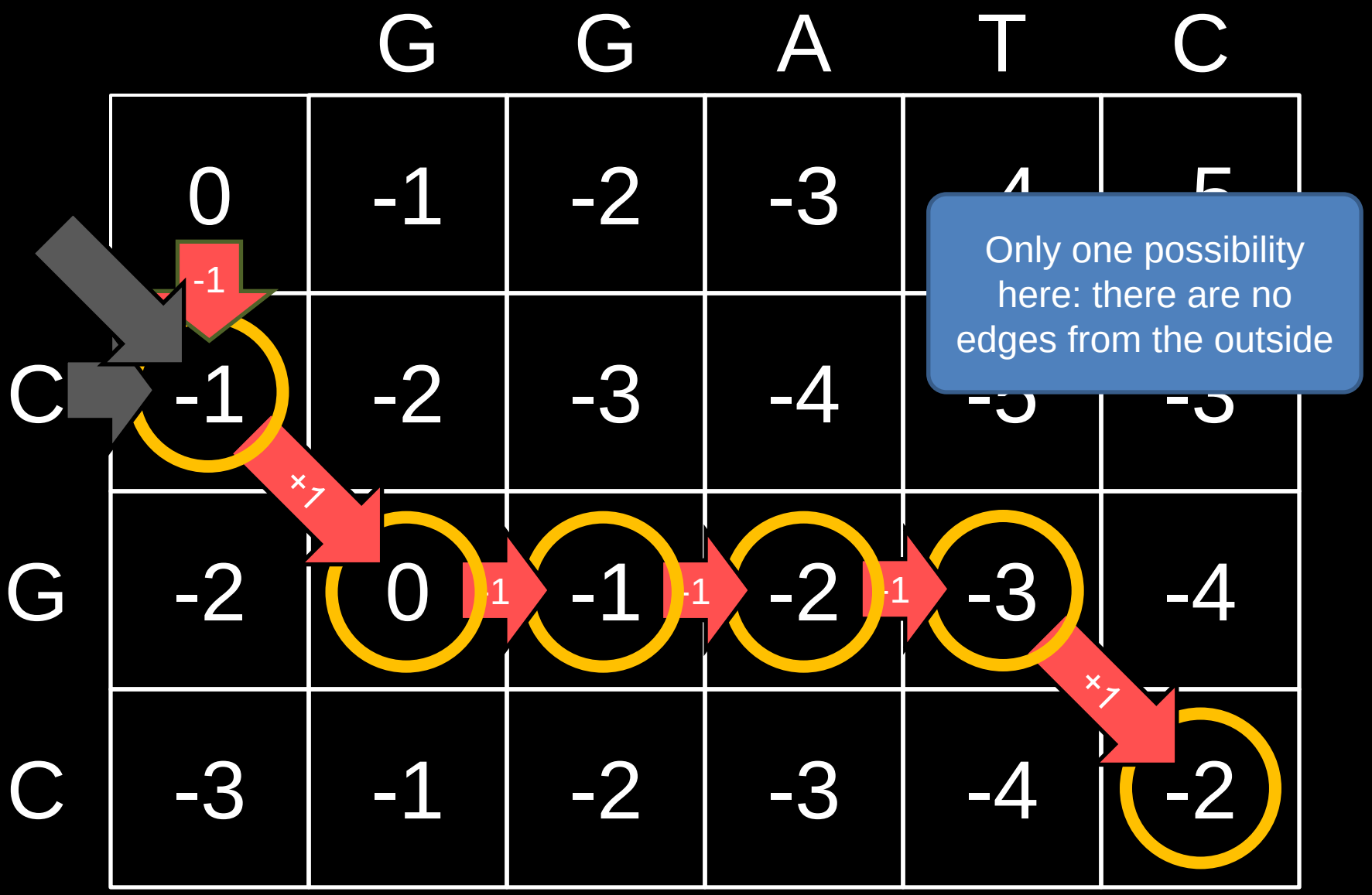
Backtracking Step 4: Find the path we took



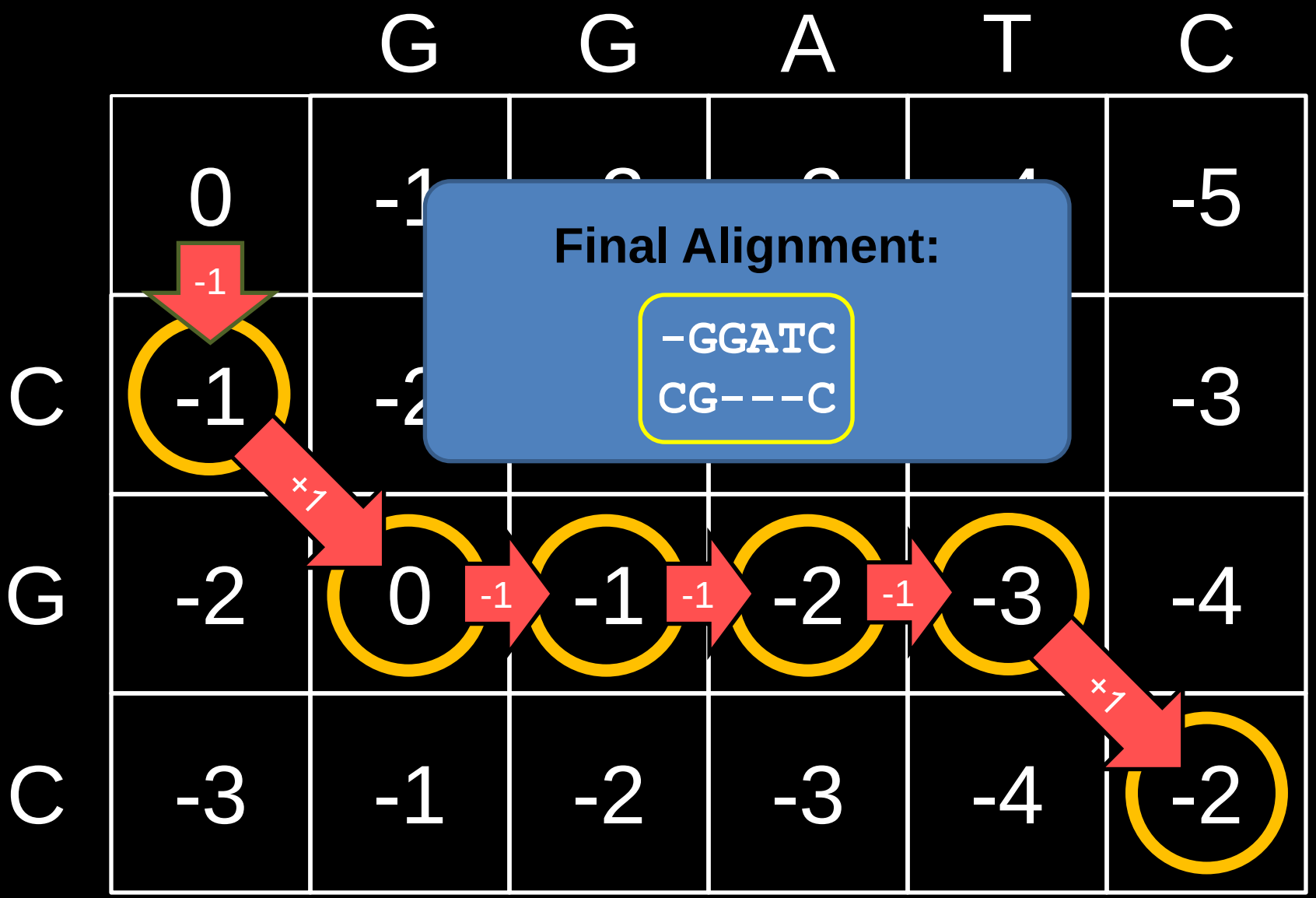
Backtracking Step 5: Find the path we took



Backtracking Step 6: Find the path we took



Backtracking Result



Next time: Affine Gap Scoring

- So far we have talked about linear gap penalties.
  - For each gap, you add a certain score (-1)
- Affine Gap penalties have a gap opening penalty as well as a gap extension penalty
  - Opening penality > extension penalty
- Affine Gap penalties encourage high scoring sequences to avoid lots of little gaps

- For next time, carefully read Jones and Pevsner Chapter 6.9. It's complicated.

Questions