Genome Rearrangements

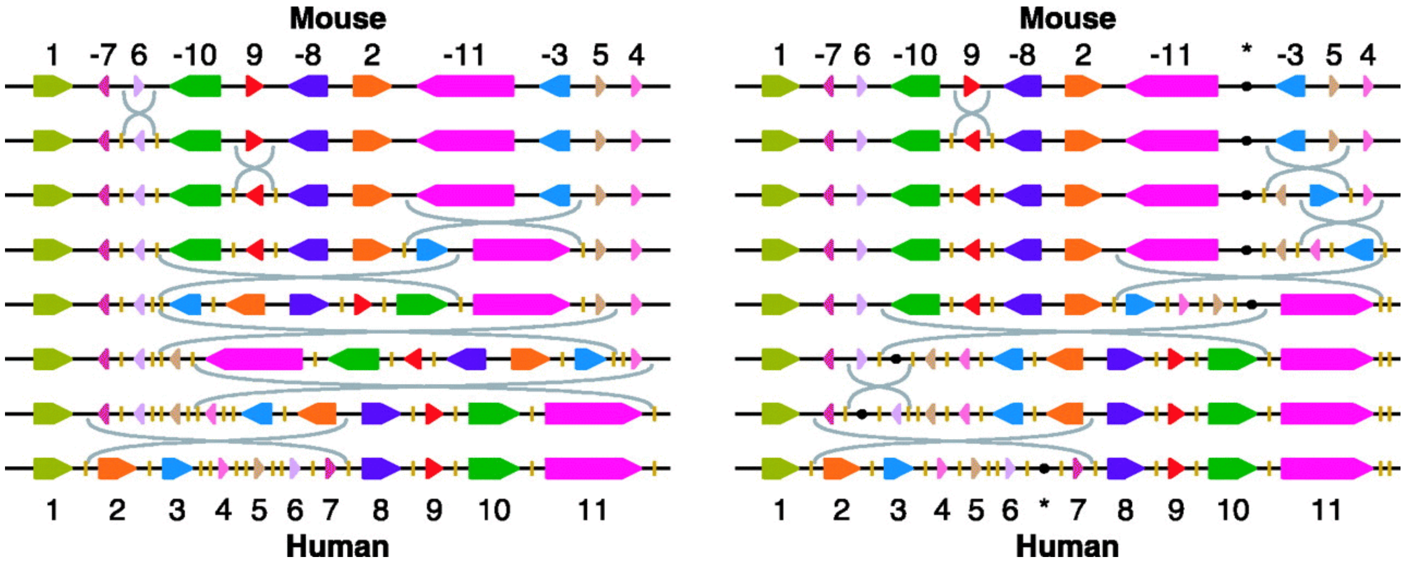


Fig. 2. Two different most parsimonious scenarios that transform the order of the 11 synteny blocks on the mouse X chromosome into the order on the human X chromosome. (...)

From: *Human and mouse genomic sequences reveal extensive breakpoint reuse in mammalian evolution* by Pavel Pevzner and Glenn Tesler. Proceedings of the National Academy of Sciences of the United States of America.

The Rearrangement Problem

Given oriented symbols:

1	–7	6	–10	9	–8	2	–11	–3	5	4

Find a set of reversals that will return it to the "home" position:

1	2	3	4	5	6	7	8	9	10	11

Each reversal:

- Selects some interval
- Reverses the order of the elements in that interval
- Changes the direction (sign) of those elements

The First Optimal Rearrangement

1	–7	6	–10	9	–8	2	–11	–3	5	4
		▨								
1	–7	–6	–10	9	–8	2	–11	–3	5	4
				▨						
1	–7	–6	–10	–9	–8	2	–11	–3	5	4
							▨	▨		
1	–7	–6	–10	–9	–8	2	3	11	5	4
			▨	▨	▨	▨	▨			
1	–7	–6	–3	–2	8	9	10	11	5	4
			▨	▨	▨	▨	▨	▨	▨	
1	–7	–6	–5	–11	–10	–9	–8	2	3	4
				▨	▨	▨	▨	▨	▨	▨
1	–7	–6	–5	–4	–3	–2	8	9	10	11
	▨	▨	▨	▨	▨	▨				
1	2	3	4	5	6	7	8	9	10	11

Four Minute Contest

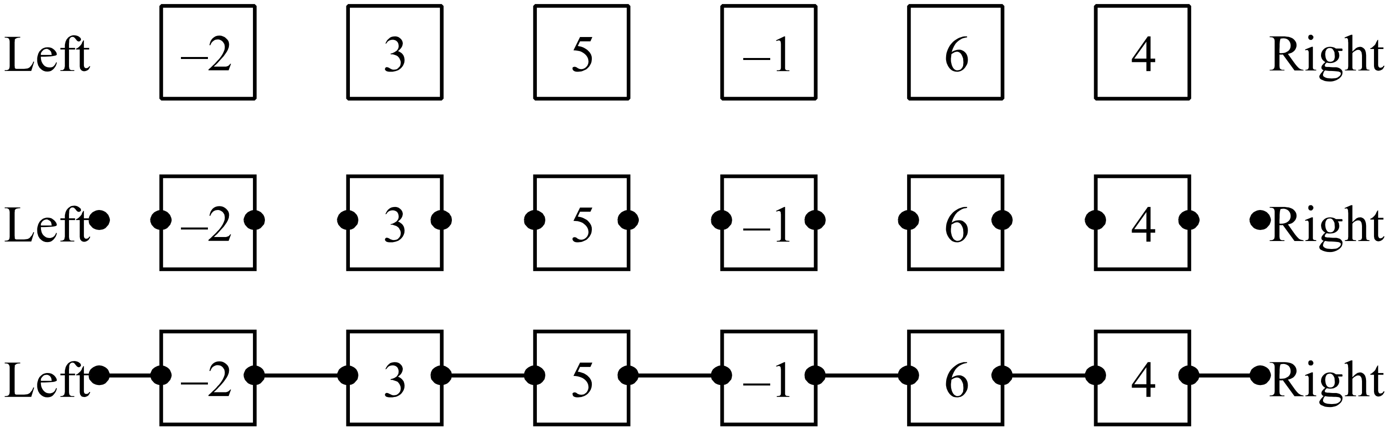
What is the least number of reversals needed to return the following oriented permutation to the home position?

–2	3	5	–1	6	4

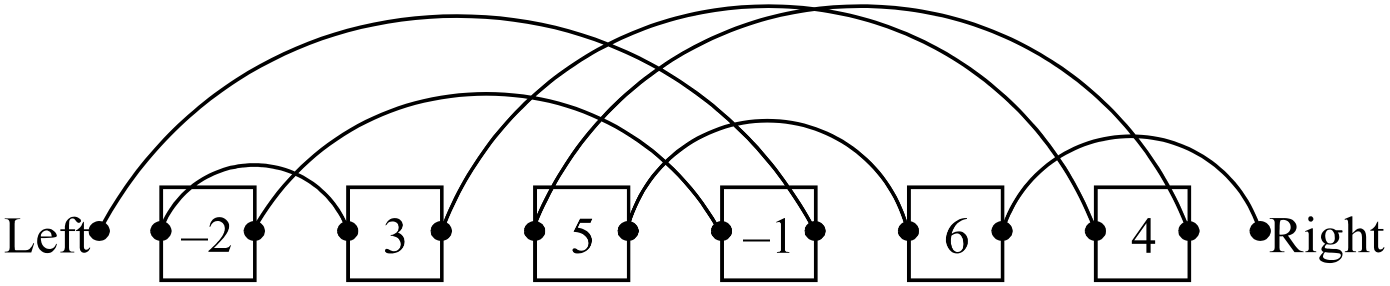
We’ll see shortly how to do this in 5 reversals, and how to find that optimal solution by hand.

The Reality and Desire Diagram

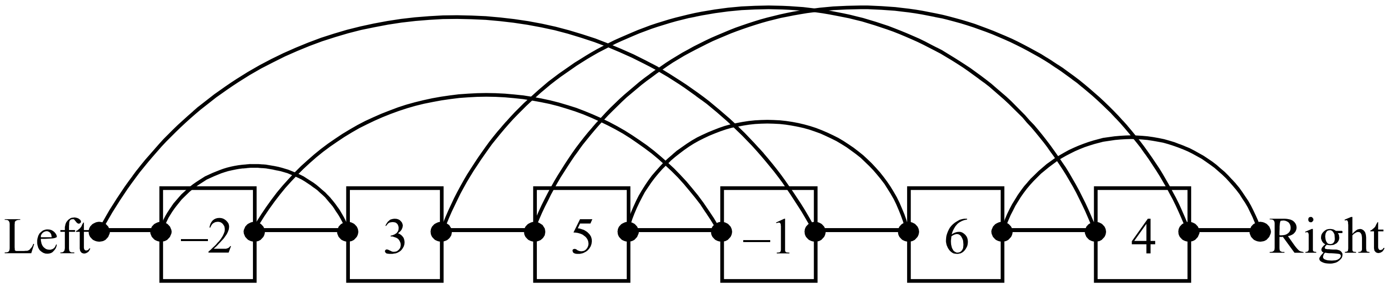
Our first step will be to draw a diagram comparing where things are with where we would like them to be.



Reality

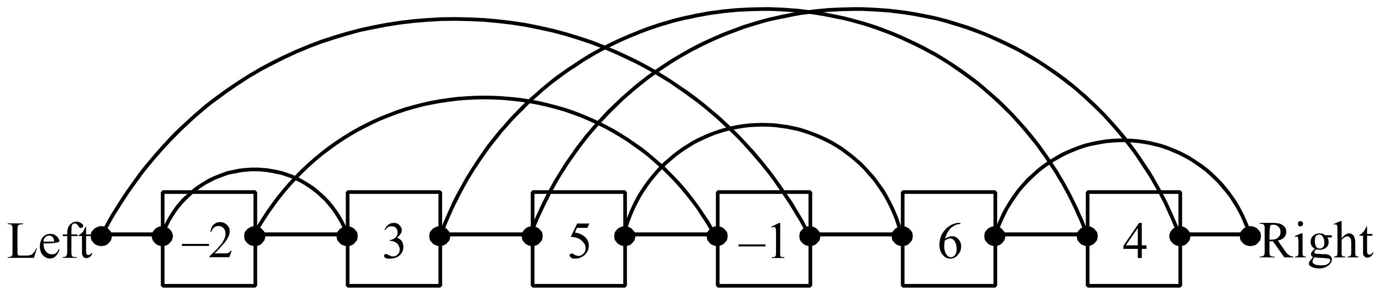


Desire



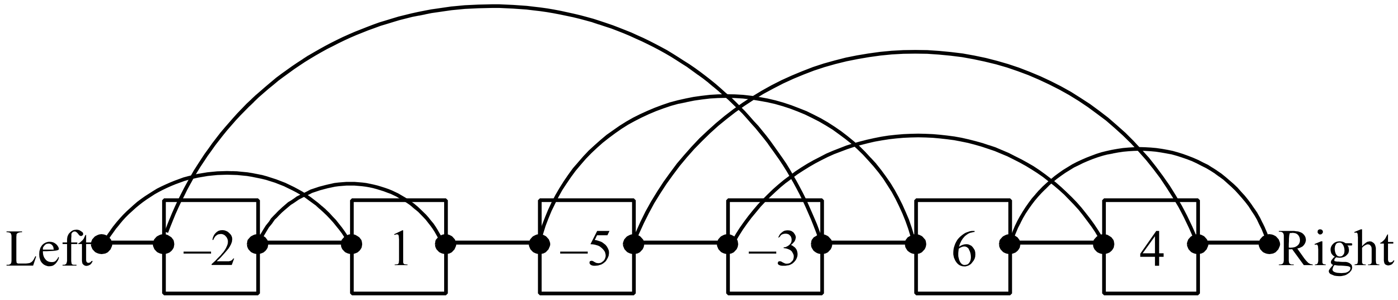
Reality and Desire

Reversals in the R and D Diagram



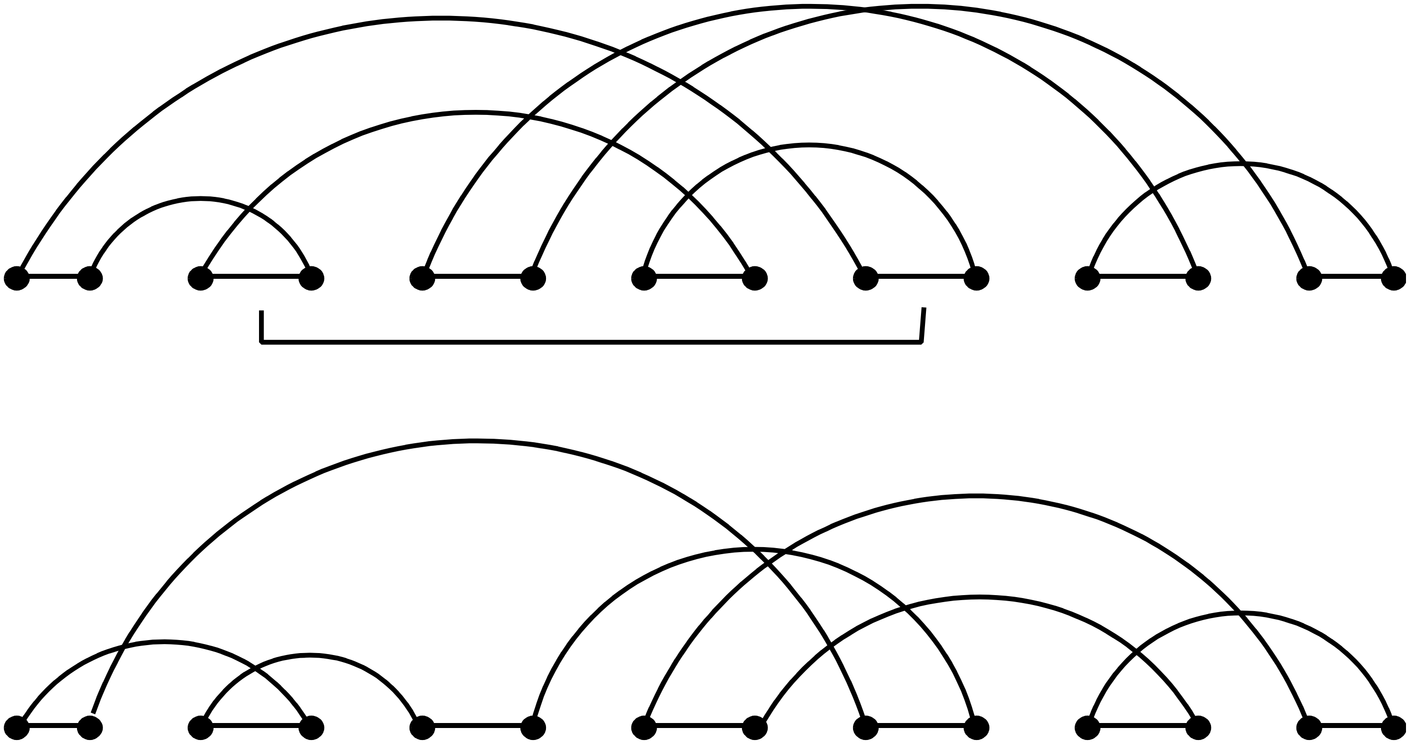
Reality and Desire

Reverse the interval [3, 5, 1]:



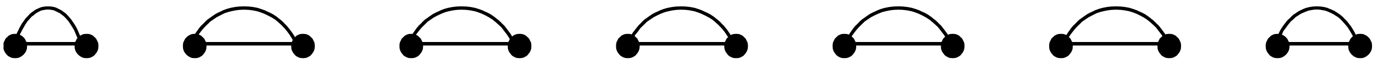
Reality and Desire

Without the Boxes and Labels



It is possible to treat these reversals as moves on a graph, independent of the permutation context.

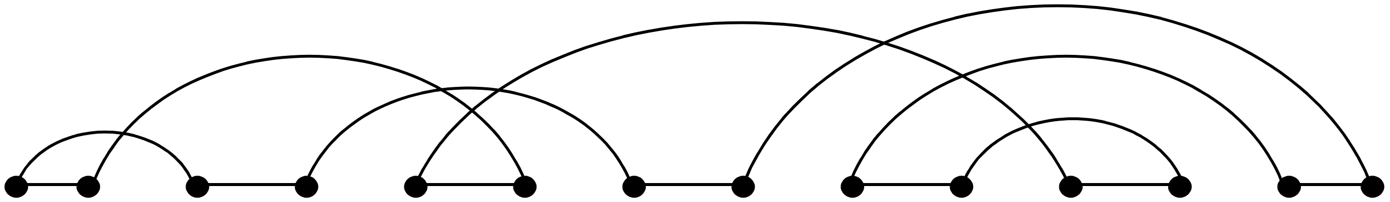
In these terms, we wish to achieve the home position:

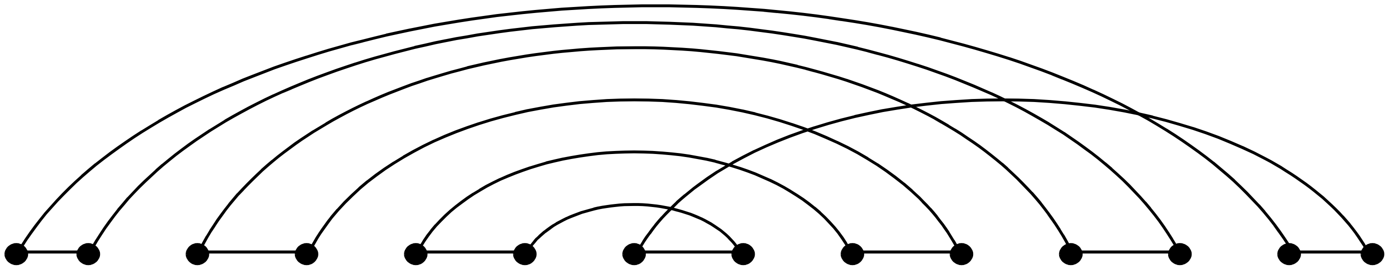


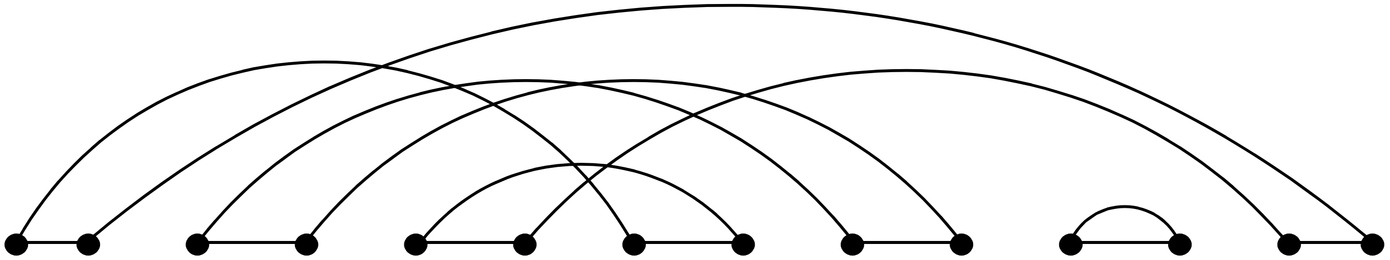
Let’s try a Handout...

Reality and Desire Diagrams

Which permutation generates each R&D diagram?







Now draw the Reality and Desire diagram for the oriented permutation “-5 1 2 -4 -6 3"

It’s All About Cycles

Consider the R&D diagram of an oriented permutation of n elements.

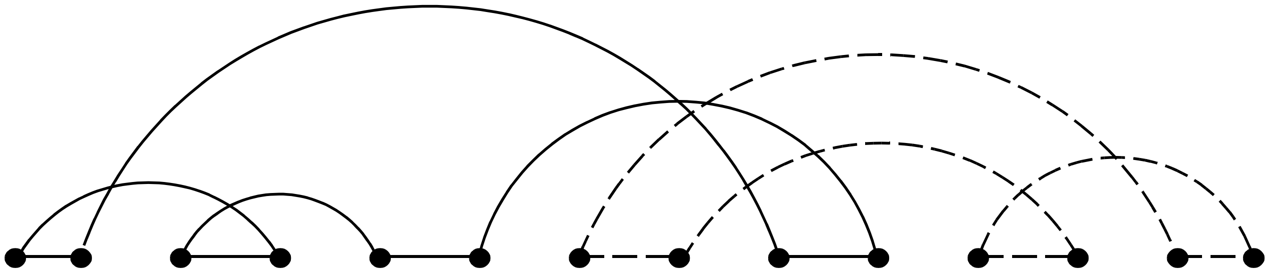
Some Facts:

- Home position has $n+1$ cycles, and is the only position with that many cycles.
- Every reversal changes the number of cycles by at most 1

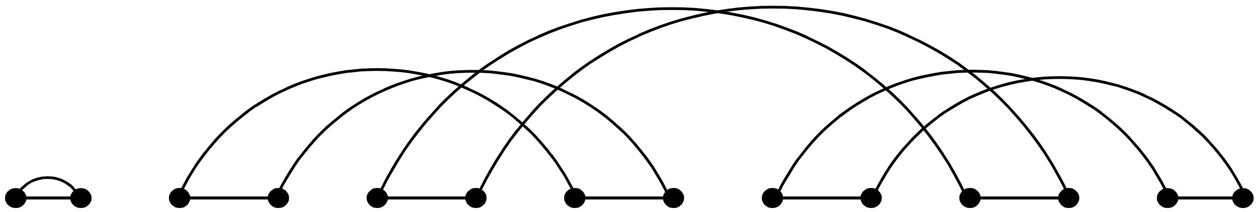
Thus if c is the number of cycles in the R&D diagram, then $n + 1 - c$ is a lower bound on the number of reversals needed to fix the permutation.

In fact, this is the correct answer for the majority of oriented permutations.

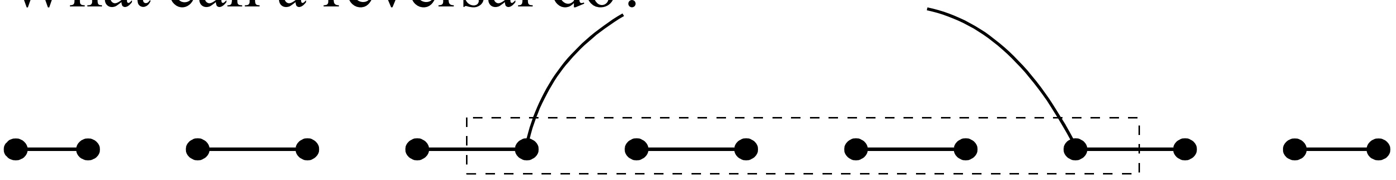
Reversals and Cycles



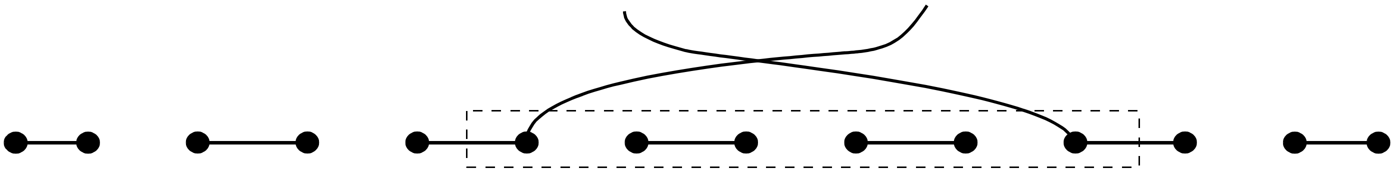
Our diagram has 2 cycles. The one below has 4:



What can a reversal do?



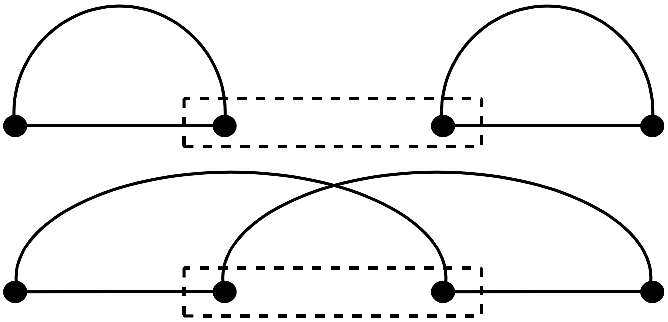
Before Reversal



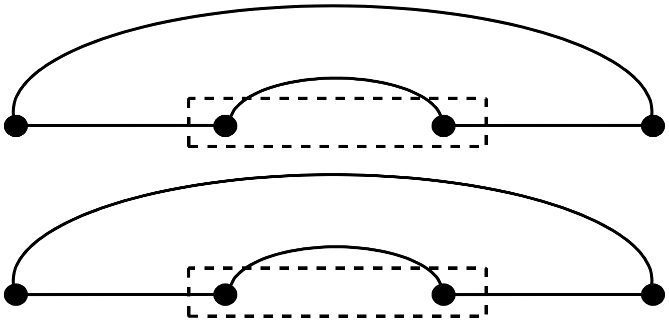
After Reversal

All other edges stay the same.

Schematically

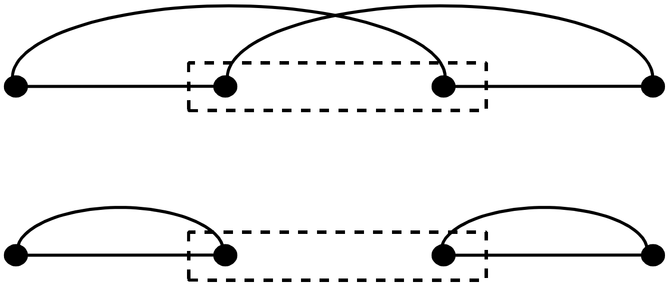


Edges in Different Cycles



Convergent edges in a cycle

Two reality edges in a cycle are said to be *convergent* if they are traversed in the same direction as one travels around the cycle. Otherwise, they are *divergent*.



Divergent edges in a cycle

Case 1: Two cycles are joined

Case 2: Two convergent reality edges are broken and the two paths are joined

Case 3: Two divergent reality edges are broken and the cycle is split into 2.

Algorithm — First Approximation

We know that we need to increase the number of cycles to $n+1$, and then we are done.

Algorithm:

At each step, find a cycle with a pair of divergent reality edges and perform the reversal that breaks those two edges.

Considerations:

- Will there always be such a pair of edges?
- Is it possible that without looking ahead we could eliminate divergent reality edges, in other cycles, that would have helped us?
- Is it possible that by looking ahead carefully we can create divergent reality edges in other cycles which otherwise would have had none?

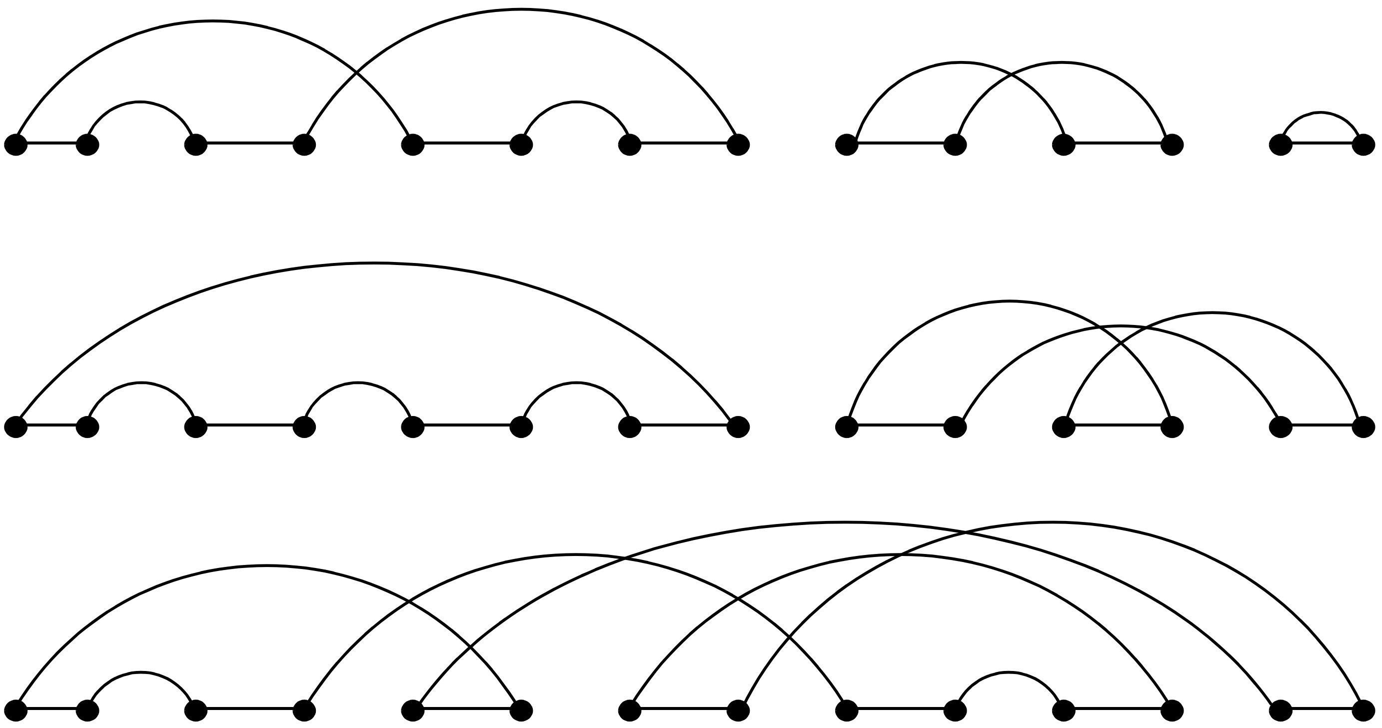
Answers:

- No
- Yes
- Yes!

Good and Bad Cycles

Call a cycle “Good” if it contains a pair of divergent edges. Otherwise, call it “Bad.”

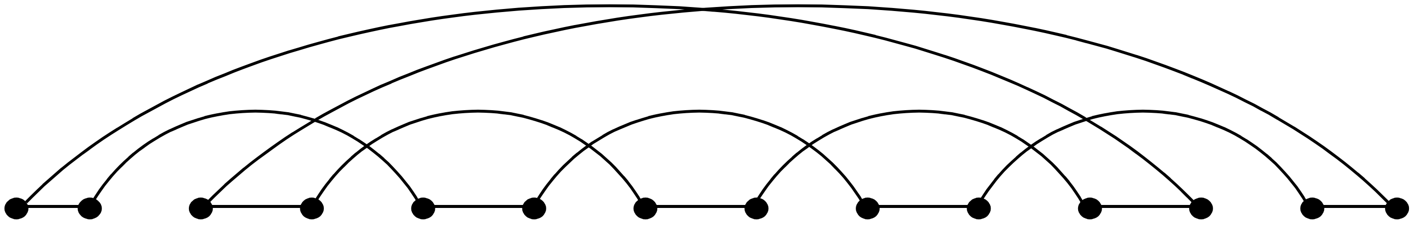
For each R&D diagram, identify the good and bad cycles.



Note: These cycles may have come from *E. coli*.

Bad Cycles

Consider the following R&D diagram:



There is just one cycle. Since home has 7 cycles, we know we need at least 6 reversals to fix this oriented permutation.

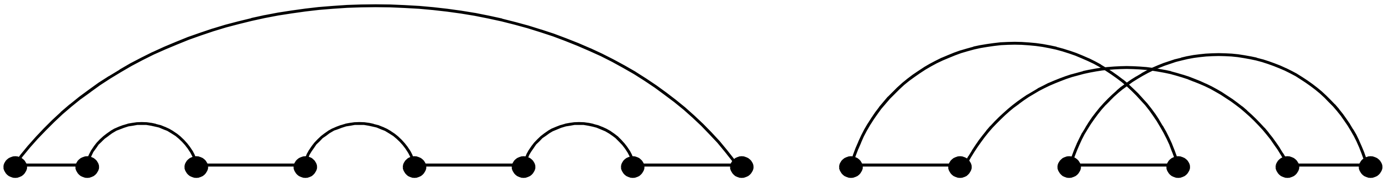
But... This one cycle is a *bad* cycle...

That means our first move cannot break this into two cycles, forcing at least one extra move.

In fact, this can be done in exactly 7 moves at best.

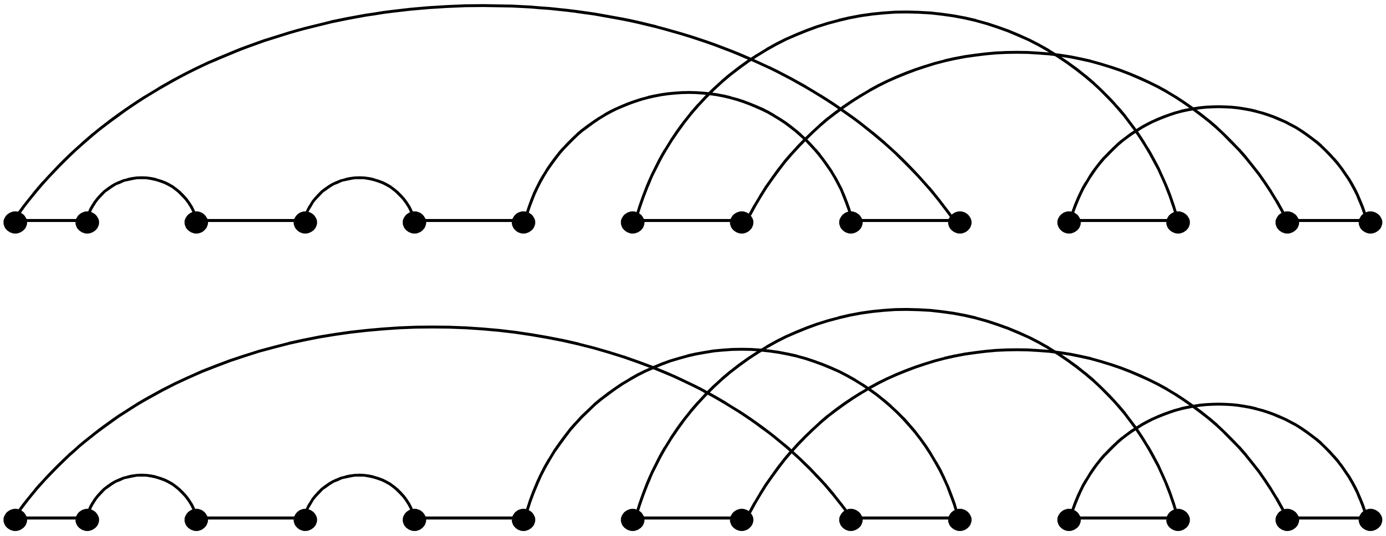
What About Two Bad Cycles?

Consider the second of our figures from the handout:



Reversals within one cycle do not affect the other cycle. Thus each cycle will need one “extra” reversal.

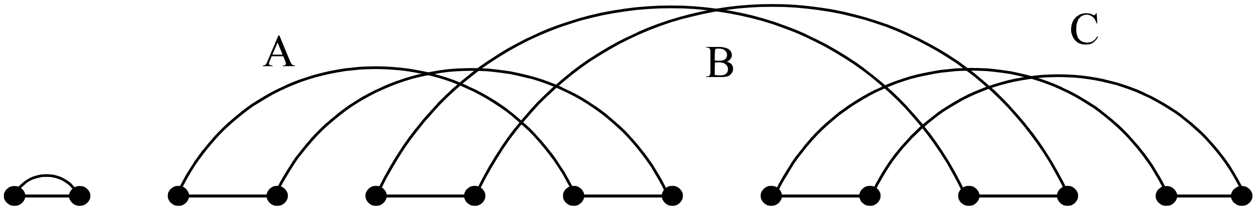
What about these cycles?



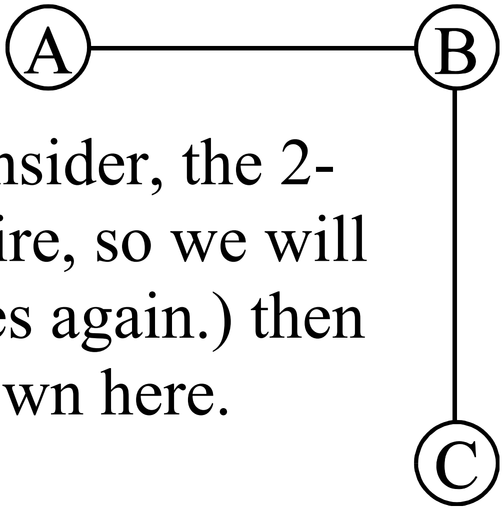
Components

The structures to look at are the components of the "overlap graph" which has a vertex for each cycle of the R&D diagram, and two vertices connected if their cycles contain intersecting edges.

The idea is that when we perform a reversal in one of the cycles, we may be able to arrange for the other to obtain and/or keep a pair of divergent edges.



If we label the cycles (Notice that we do not label, or even consider, the 2-cycle. There, reality equals desire, so we will not have to deal with those edges again.) then we obtain the overlap graph shown here.



Good Components and Bad Components

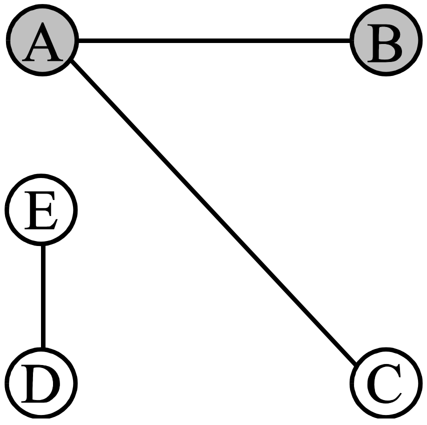
A component of a graph is a maximal connected subset of its vertices.

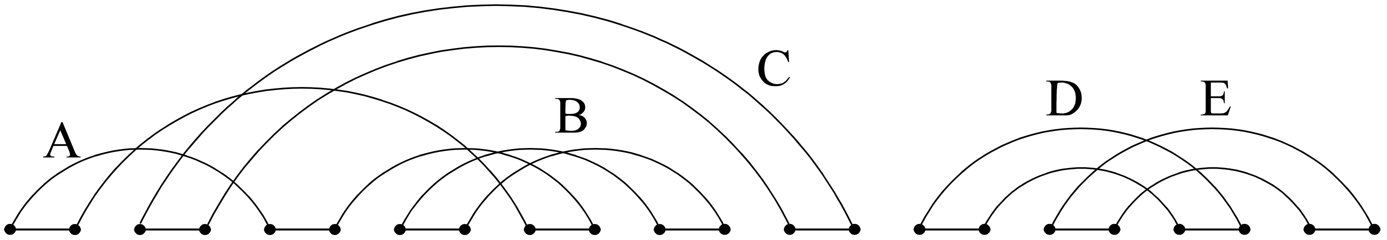
A component is called "good" if it contains any good cycles. Otherwise it is bad.

In the previous example all three cycles were good, so the component was good.

In the example below there are 5 cycles. Their overlap graph is shown below; shaded vertices correspond to good cycles.

The "ABC" component is a good component, while the "DE" component is bad.

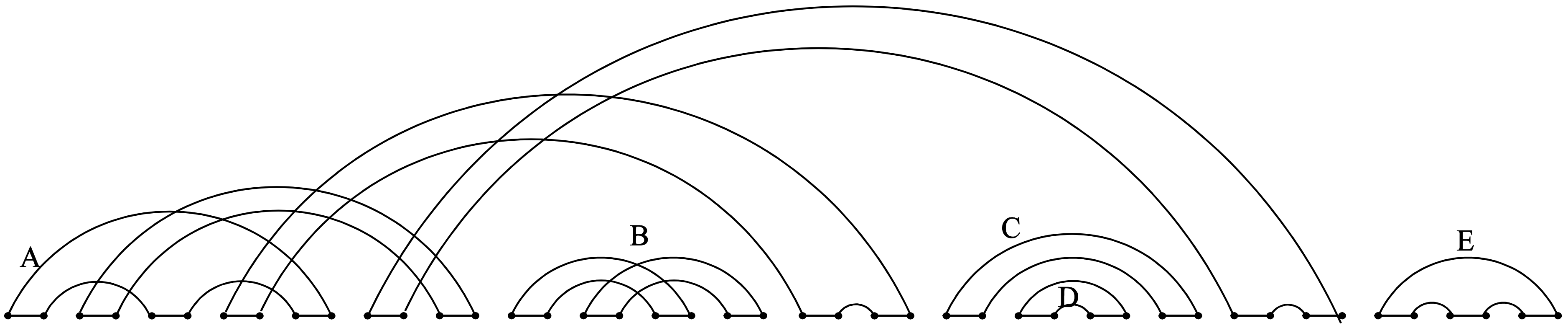




Hurdles

Hurdles are bad components which do not separate any two other bad components. These are the only bad components which contribute to the number of reversals needed, since non-hurdle bad components can be "twisted" to obtain a divergent pair.

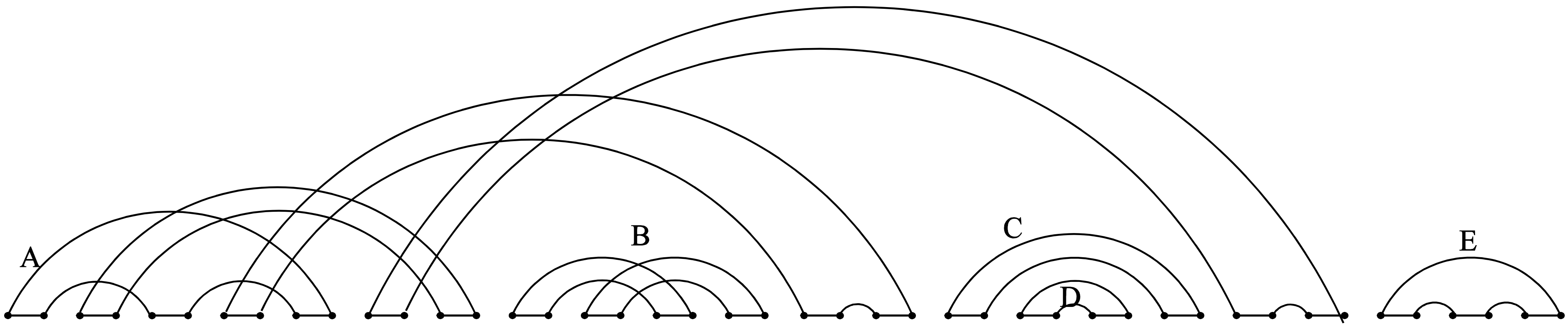
The example on the previous slide had 1 hurdle.



The hurdles here are B, D and E.

Protection and Super Hurdles

A Hurdle A is said to *protect* a non-hurdle, bad component B, if removal of A causes B to become a hurdle.
Who protects whom here?



D protects C, and that’s the only case.

A hurdle protecting a non-hurdle is called a *super hurdle.*

Fortresses

A *Fortress* is a permutation whose reality and desire diagram contains an odd number of hurdles, all of which are super hurdles. These contribute one more to the number of reversals we need to fix the permutation.

The Formula

Minimum number of reversals needed is:

$$\boxed{n + 1 - \#\text{ of cycles} + \#\text{ of hurdles} + \text{F}}$$

Where F = 1 if the permutation is a fortress, and 0 otherwise.

The Algorithm

On each step:

- If there is a good component, find a pair of divergent edges in some cycle in that component and reverse the interval between them. Be sure, however, that after the reversal the component is still good. (This is always possible.)

- If there are only bad components, try to find a bad component which is not a hurdle. Find hurdles on each side of the component, select an edge from each and reverse the interval between those edges.

- If only hurdles remain, reverse the interval between any pair of (convergent) edges in that hurdle, but not in a super hurdle if possible.

Details remain. See Setubal and Meidanis.

Handout #1 — Sorting by Reversals

Here are six copies of an oriented permutation. What is the least number of reversals needed to restore it to its proper orientation "+1 +2 +3 +4 +5 +6"?

–2	3	5	–1	6	4

–2	3	5	–1	6	4

–2	3	5	–1	6	4

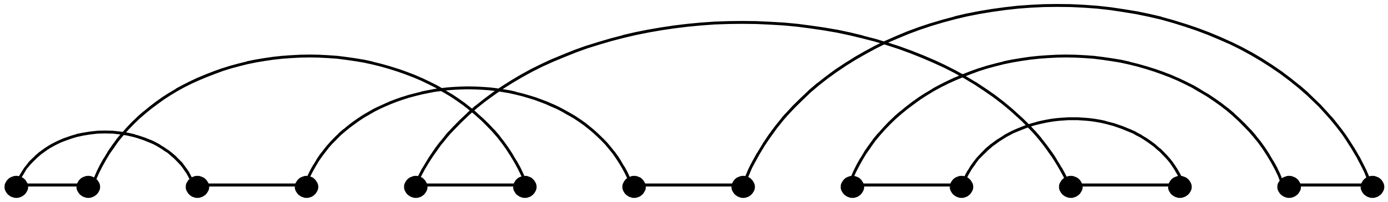
–2	3	5	–1	6	4

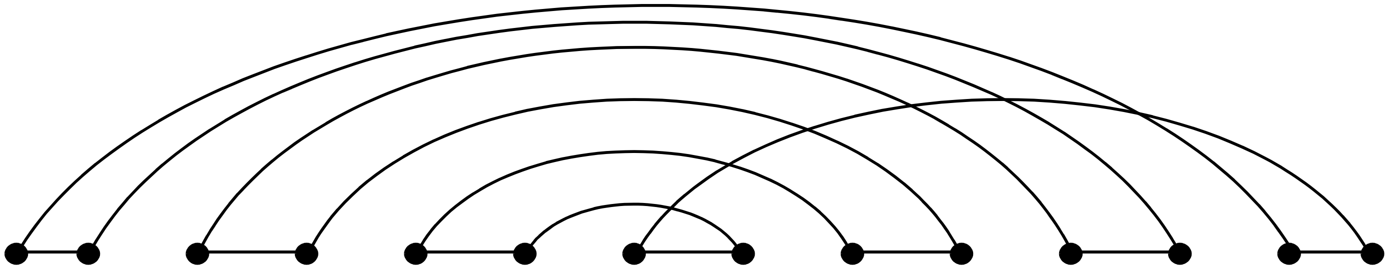
–2	3	5	–1	6	4

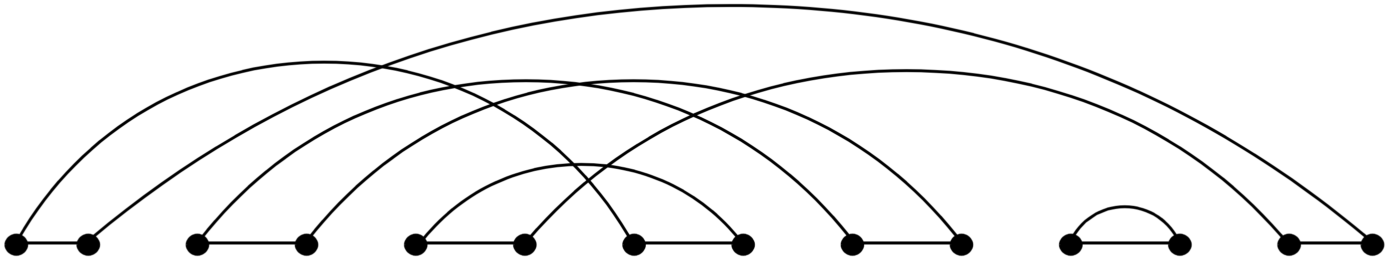
–2	3	5	–1	6	4

Handout #2 — Oriented Permutations and Reality and Desire Diagrams

For each of the Reality and Desire diagrams below, identify the oriented permutation that generated it.



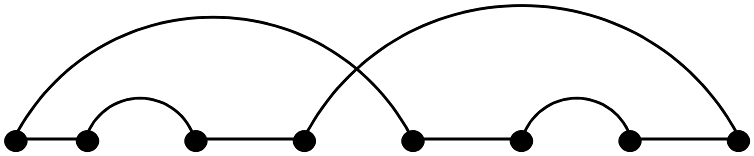


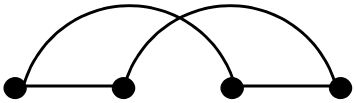


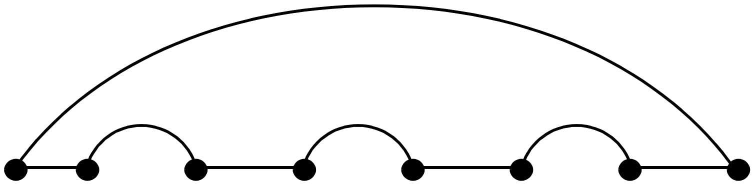
Now draw the Reality and Desire diagram for the oriented permutation “-5 1 2 -4 -6 3"

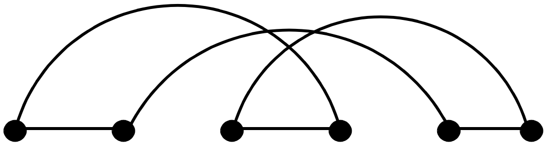
Handout #3 — Good Cycles and Bad Cycles

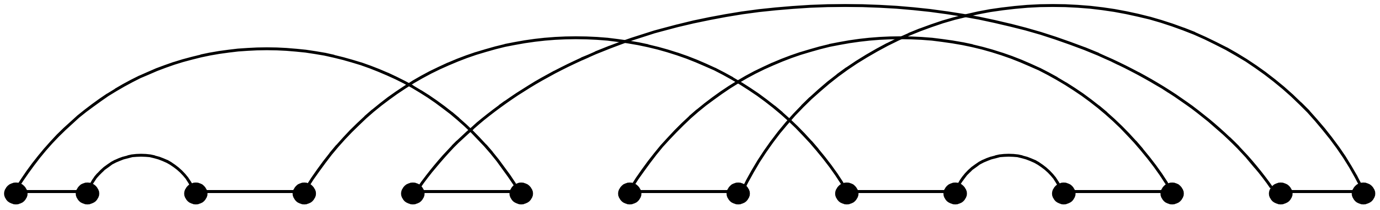
In each of the reality and desire diagrams below, identify the good and bad cycles. For those cycles which you identify as “good,” show a pair of divergent edges. [*Note: These diagrams were generated to be simple examples of R&D cycle structure. They do not arise from actual oriented permutations.*]











Exercise — Genome Rearrangement

The two matrices are alignments of two sequences according to the Smith-Waterman algorithm. The first matrix has both sequences in the “forward” direction. The second has sequence 2 (the one down the first column) reversed.

Your job is to find out how to split the first sequence into a few pieces (the fewer, the better) so that they can be rearranged to make the second sequence in the same way that the mouse X chromosome could be rearranged to make the human X chromosome (see the first slide in your notes).

Finally, once you have identified the pieces (there are about 5 of them) find the optimal sequence of reversals to transform the first sequence into the second.

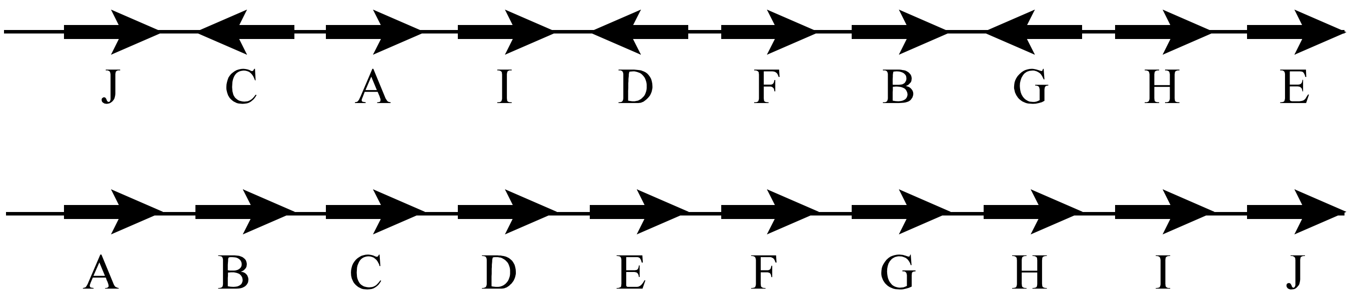
Exercises — Genome Rearrangement

Presentation Problems:

1. Draw a Fortress. That is, a reality and desire diagram which has an odd number of hurdles, all of which are super hurdles. (Note that it does not need to arise from an actual signed permutation.)

2. What is the least number of vertices possible in a reality and desire diagram which contains a super hurdle? (Note that it does not need to arise from an actual signed permutation.)

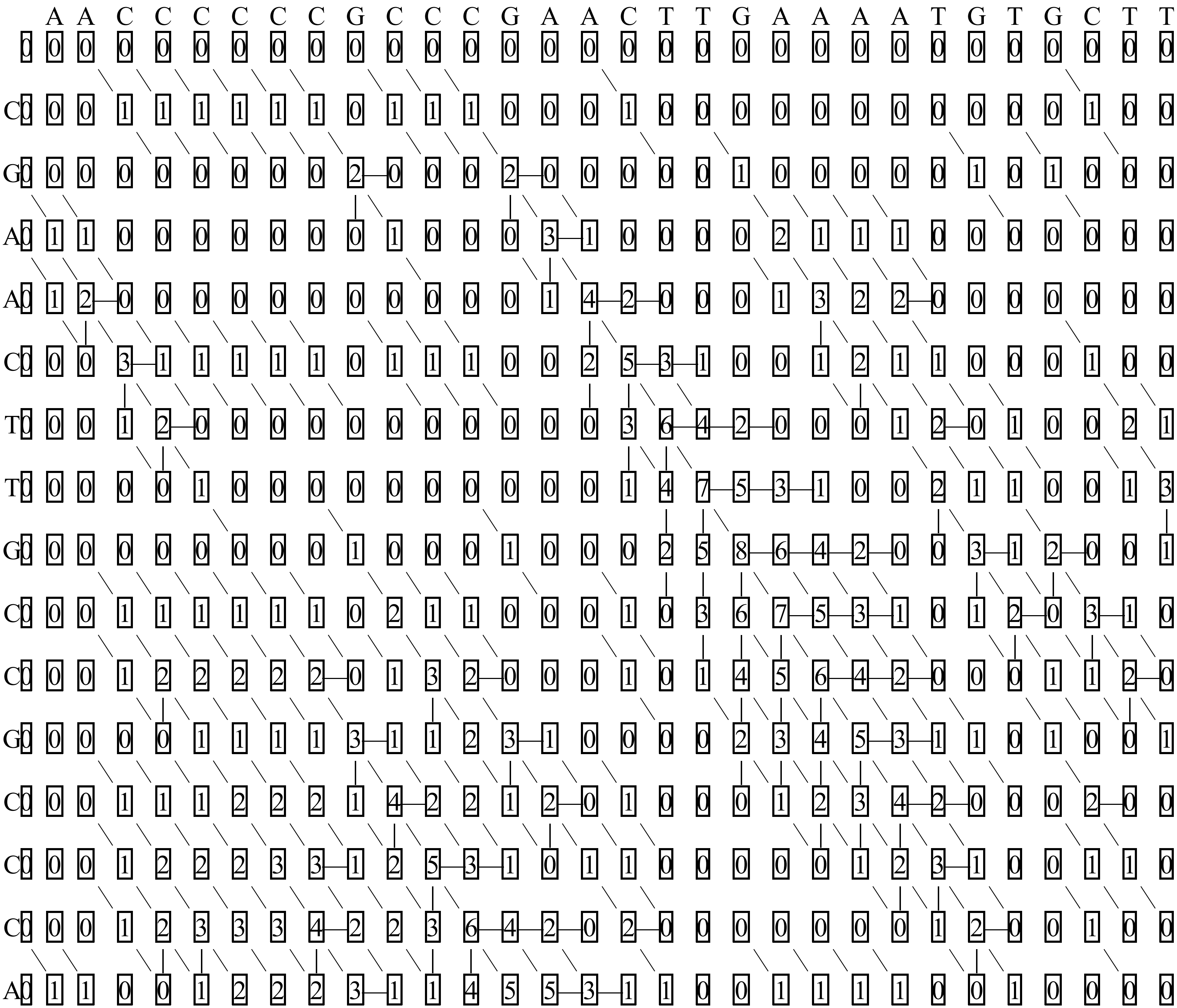
3. The figure below represents a breakdown of two genomes into ten pieces which align well. What is the least number of reversals necessary to explain this transformation? What reversals will do the trick? (Note that these arrows refer merely to the direction of the DNA fragment represented by the letter, and are not meant to be the directions of *any* reality or desire edge



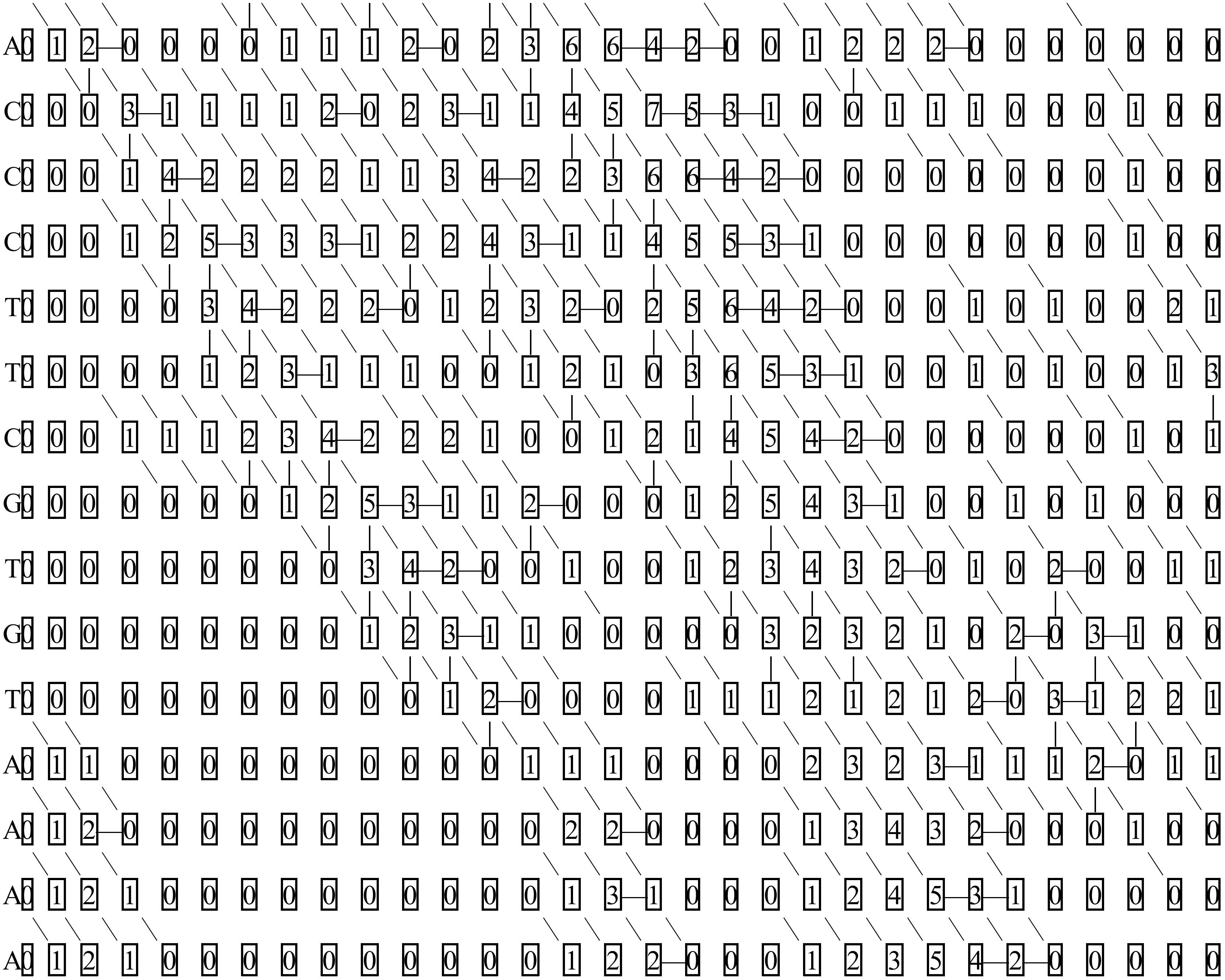
4. Suppose that the arrow above the "G" was reversed in the top genome above. What would be "wrong" with that description of the way the two genomes were related?

5. Find an applet (or other program) online which does sorting by reversals. Generate 40 random permutations, and use this program to compute the number of your random permutations which contain
   a. hurdles
   b. super hurdles
   c. fortresses.

   What is the lesson when it comes to "real life" permutations?

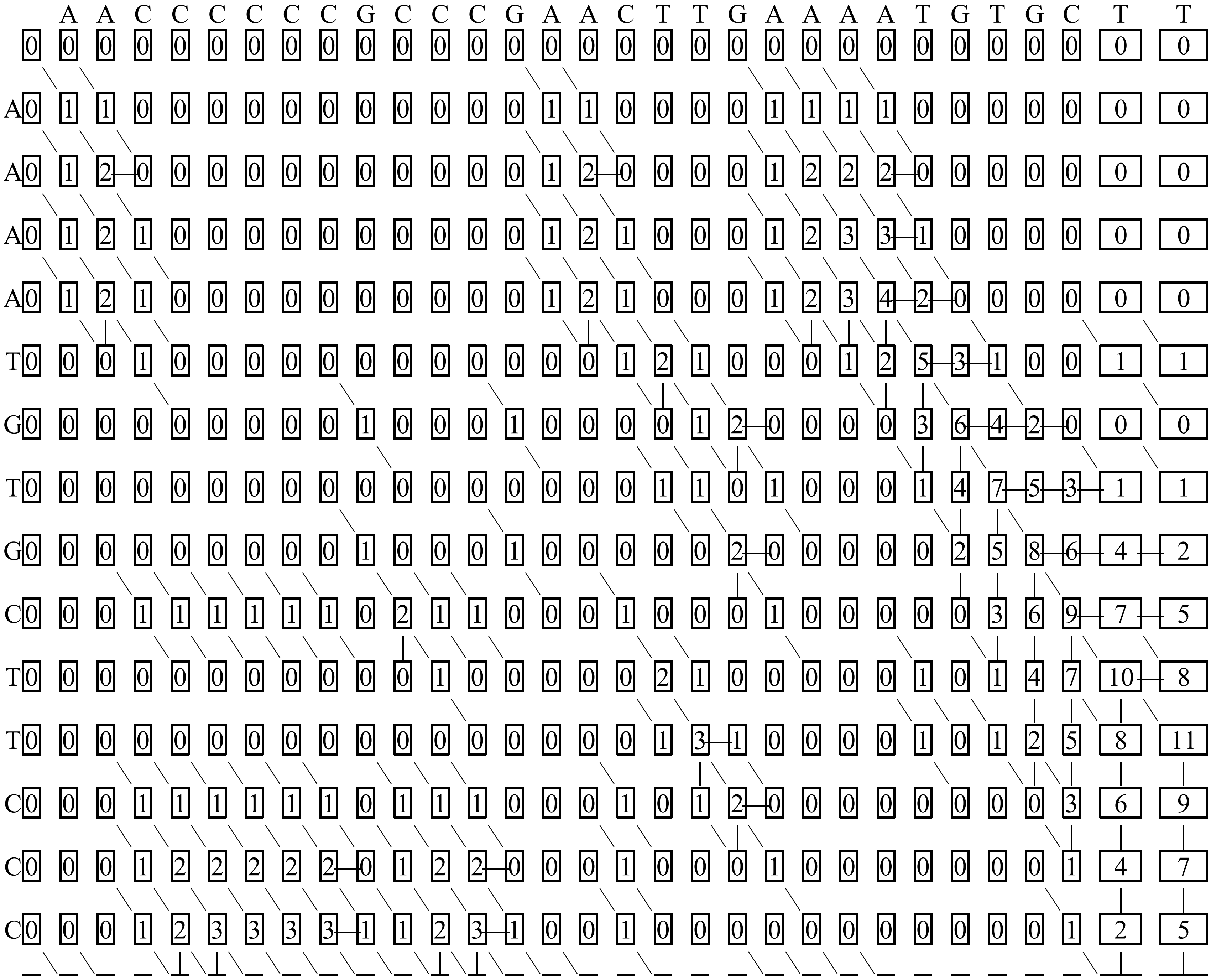


		A	A	C	C	C	C	C	C	G	C	C	C	G	A	A	C	T	T	G	A	A	A	A	T	G	T	G	C	T	T
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	0	1	1	1	1	1	1	0	1	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
G	0	0	0	0	0	0	0	0	0	2	0	0	0	2	0	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0
A	0	1	1	0	0	0	0	0	0	0	1	0	0	0	3	1	0	0	0	0	2	1	1	1	0	0	0	0	0	0	0
A	0	1	2	0	0	0	0	0	0	0	0	0	0	0	1	4	2	0	0	0	1	3	2	2	0	0	0	0	0	0	0
C	0	0	0	3	1	1	1	1	1	0	1	1	1	0	0	2	5	3	1	0	0	1	2	1	1	0	0	0	1	0	0
T	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0	0	3	6	4	2	0	0	0	1	2	0	1	0	0	2	1
T	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	4	7	5	3	1	0	0	2	1	1	0	0	1	3
G	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	2	5	8	6	4	2	0	0	3	1	2	0	0	1
C	0	0	0	1	1	1	1	1	1	0	2	1	1	0	0	0	1	0	3	6	7	5	3	1	0	1	2	0	3	1	0
C	0	0	0	1	2	2	2	2	2	0	1	3	2	0	0	0	1	0	1	4	5	6	4	2	0	0	0	1	1	2	0
G	0	0	0	0	0	1	1	1	1	3	1	1	2	3	1	0	0	0	0	2	3	4	5	3	1	1	0	1	0	0	1
C	0	0	0	1	1	1	2	2	2	1	4	2	2	1	2	0	1	0	0	0	1	2	3	4	2	0	0	0	2	0	0
C	0	0	0	1	2	2	2	3	3	1	2	5	3	1	0	1	1	0	0	0	0	0	1	2	3	1	0	0	1	1	0
C	0	0	0	1	2	3	3	3	4	2	2	3	6	4	2	0	2	0	0	0	0	0	0	0	1	2	0	0	1	0	0
A	0	1	1	0	0	1	2	2	2	3	1	1	4	5	5	3	1	1	0	0	1	1	1	1	0	0	1	0	0	0	0

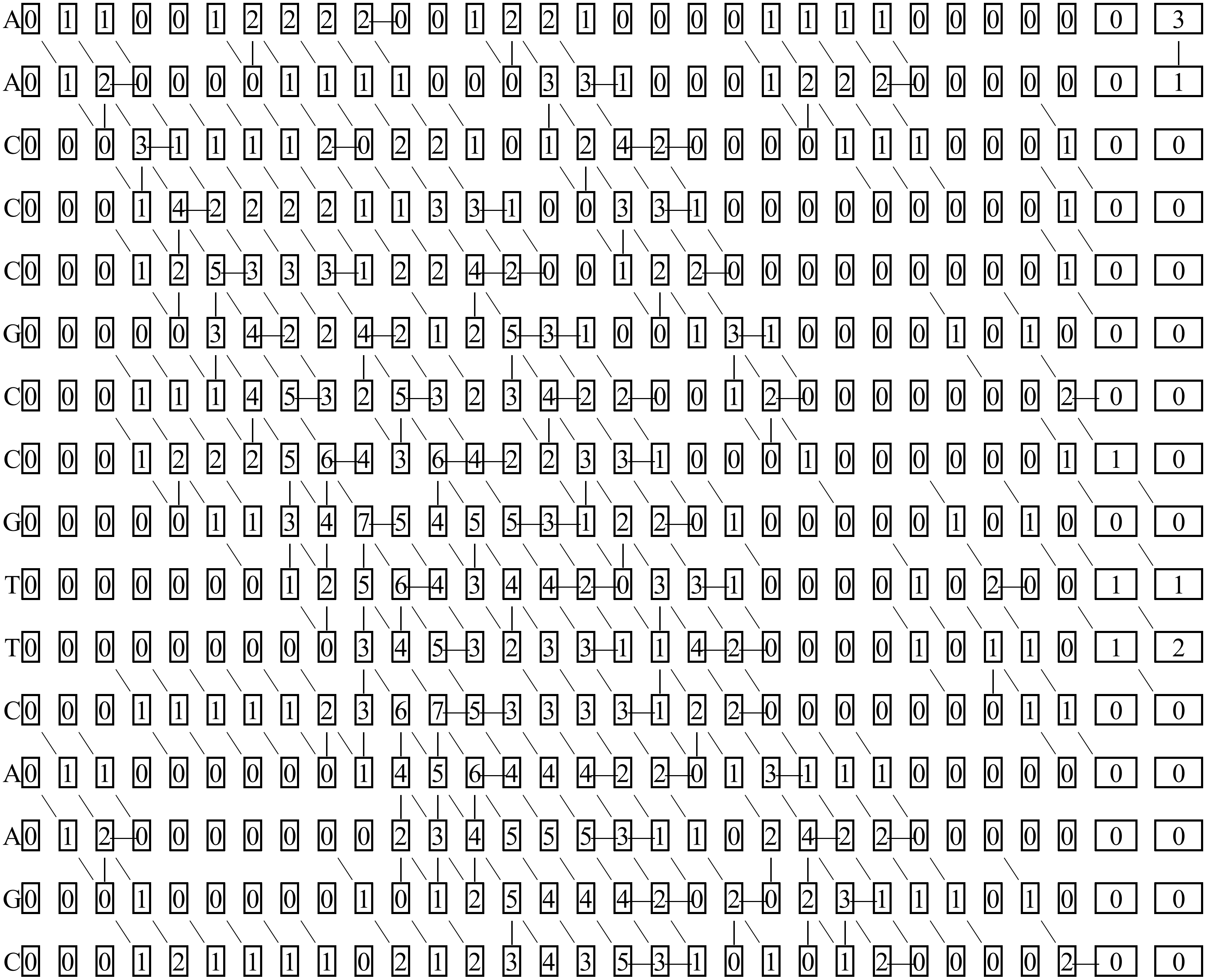


A	0	1	2	0	0	0	0	1	1	1	2	0	2	3	6	6	4	2	0	0	1	2	2	2	0	0	0	0	0	0	0
C	0	0	0	3	1	1	1	1	2	0	2	3	1	1	4	5	7	5	3	1	0	0	1	1	1	0	0	0	1	0	0
C	0	0	0	1	4	2	2	2	2	1	1	3	4	2	2	3	6	6	4	2	0	0	0	0	0	0	0	0	1	0	0
C	0	0	0	1	2	5	3	3	3	1	2	2	4	3	1	1	4	5	5	3	1	0	0	0	0	0	0	0	1	0	0
T	0	0	0	0	0	3	4	2	2	2	0	1	2	3	2	0	2	5	6	4	2	0	0	0	1	0	1	0	0	2	1
T	0	0	0	0	0	1	2	3	1	1	1	0	0	1	2	1	0	3	6	5	3	1	0	0	1	0	1	0	0	1	3
C	0	0	0	1	1	1	2	3	4	2	2	2	1	0	0	1	2	1	4	5	4	2	0	0	0	0	0	0	1	0	1
G	0	0	0	0	0	0	0	1	2	5	3	1	1	2	0	0	0	1	2	5	4	3	1	0	0	1	0	1	0	0	0
T	0	0	0	0	0	0	0	0	0	3	4	2	0	0	1	0	0	1	2	3	4	3	2	0	1	0	2	0	0	1	1
G	0	0	0	0	0	0	0	0	0	1	2	3	1	1	0	0	0	0	0	3	2	3	2	1	0	2	0	3	1	0	0
T	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	1	1	1	2	1	2	1	2	0	3	1	2	2	1
A	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	2	3	2	3	1	1	1	2	0	1	1
A	0	1	2	0	0	0	0	0	0	0	0	0	0	0	2	2	0	0	0	0	1	3	4	3	2	0	0	0	1	0	0
A	0	1	2	1	0	0	0	0	0	0	0	0	0	0	1	3	1	0	0	0	1	2	4	5	3	1	0	0	0	0	0
A	0	1	2	1	0	0	0	0	0	0	0	0	0	0	1	2	2	0	0	0	1	2	3	5	4	2	0	0	0	0	0

Here is the alignment of the first sequence (across the top) with the second sequence reversed (down the left column).



		A	A	C	C	C	C	C	C	G	C	C	C	G	A	A	C	T	T	G	A	A	A	A	T	G	T	G	C	T	T
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0
A	0	1	2	0	0	0	0	0	0	0	0	0	0	0	1	2	0	0	0	0	1	2	2	2	0	0	0	0	0	0	0
A	0	1	2	1	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	1	2	3	3	1	0	0	0	0	0	0
A	0	1	2	1	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	1	2	3	4	2	0	0	0	0	0	0
T	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	2	1	0	0	0	1	2	5	3	1	0	0	1	1
G	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	2	0	0	0	0	3	6	4	2	0	0	0
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	1	4	7	5	3	1	1
G	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	2	0	0	0	0	0	2	5	8	6	4	2
C	0	0	0	1	1	1	1	1	1	0	2	1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	3	6	9	7	5
T	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2	1	0	0	0	0	0	1	0	1	4	7	10	8
T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	3	1	0	0	0	0	1	0	1	2	5	8	11
C	0	0	0	1	1	1	1	1	1	0	1	1	1	0	0	0	1	0	1	2	0	0	0	0	0	0	0	0	3	6	9
C	0	0	0	1	2	2	2	2	2	0	1	2	2	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	4	7
C	0	0	0	1	2	3	3	3	3	1	1	2	3	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	2	5



A	0	1	1	0	0	1	2	2	2	2	0	0	1	2	2	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0	3
A	0	1	2	0	0	0	0	1	1	1	1	0	0	0	3	3	1	0	0	0	1	2	2	2	0	0	0	0	0	0	1
C	0	0	0	3	1	1	1	1	2	0	2	2	1	0	1	2	4	2	0	0	0	0	1	1	1	0	0	0	1	0	0
C	0	0	0	1	4	2	2	2	2	1	1	3	3	1	0	0	3	3	1	0	0	0	0	0	0	0	0	0	1	0	0
C	0	0	0	1	2	5	3	3	3	1	2	2	4	2	0	0	1	2	2	0	0	0	0	0	0	0	0	0	1	0	0
G	0	0	0	0	0	3	4	2	2	4	2	1	2	5	3	1	0	0	1	3	1	0	0	0	0	1	0	1	0	0	0
C	0	0	0	1	1	1	4	5	3	2	5	3	2	3	4	2	2	0	0	1	2	0	0	0	0	0	0	0	2	0	0
C	0	0	0	1	2	2	2	5	6	4	3	6	4	2	2	3	3	1	0	0	0	1	0	0	0	0	0	0	1	1	0
G	0	0	0	0	0	1	1	3	4	7	5	4	5	5	3	1	2	2	0	1	0	0	0	0	0	1	0	1	0	0	0
T	0	0	0	0	0	0	0	1	2	5	6	4	3	4	4	2	0	3	3	1	0	0	0	0	1	0	2	0	0	1	1
T	0	0	0	0	0	0	0	0	0	3	4	5	3	2	3	3	1	1	4	2	0	0	0	0	1	0	1	1	0	1	2
C	0	0	0	1	1	1	1	1	2	3	6	7	5	3	3	3	3	1	2	2	0	0	0	0	0	0	0	1	1	0	0
A	0	1	1	0	0	0	0	0	0	1	4	5	6	4	4	4	2	2	0	1	3	1	1	1	0	0	0	0	0	0	0
A	0	1	2	0	0	0	0	0	0	0	2	3	4	5	5	5	3	1	1	0	2	4	2	2	0	0	0	0	0	0	0
G	0	0	0	1	0	0	0	0	0	1	0	1	2	5	4	4	4	2	0	2	0	2	3	1	1	1	0	1	0	0	0
C	0	0	0	1	2	1	1	1	1	0	2	1	2	3	4	3	5	3	1	0	1	0	1	2	0	0	0	0	2	0	0