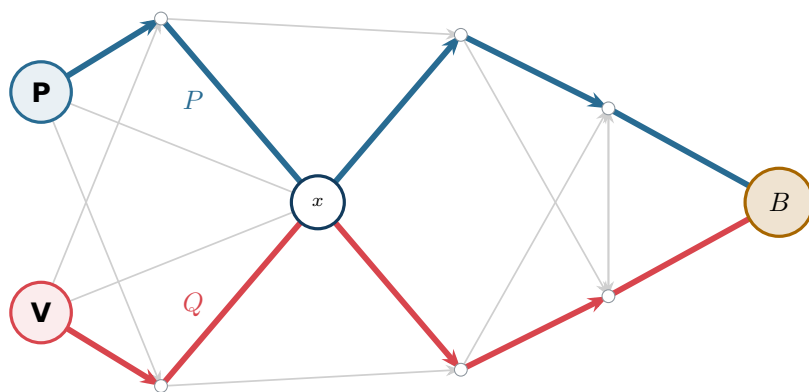


LKTG: Games on Changing Graphs

Solutions and Open Problems

Five starred parts are intentionally left unsolved



$$S = |P| + |Q| + \varepsilon$$

Prepared from canonical project version 21 and its two addenda.

Every lemma used below is proved in the text, and every cross-reference points to an earlier part of this collection. The diagrams were drawn in TikZ; the solution to Problem 6(b) includes a frame from a step-by-step demonstration of the strategy.

How to read this collection

The point is not merely to get the right answer, but to identify the game idea: a pairing strategy, an invariant, a weight, or a symmetry. Long arguments are therefore broken into short lemmas, and at difficult points we explain why the next move is useful.

Parts 11(g), 11(h), 13(d), 14(f), and Problem 27 are starred and intentionally left unsolved; their complete statements are retained. Every other part is proved without treating an external theorem as a black box.

Numbering. Problems 1–17 correspond to canonical project version 21, and Problems 18–27 to Addendum 1, version 6. The two problems from the separate 8 August sheet, “Rock, Paper, Scissors” and “Two Leagues”, are numbered 28 and 29. Thus Problems 25–27 here always belong to the weighted Erdős-game sequence.

Notation and terminology. Unless stated otherwise, all graphs are finite and simple. We write $\omega(G)$ and $\alpha(G)$ for the clique number and independence number of G , respectively; $d_G(v)$ for the degree of a vertex; $\delta(G)$ for minimum degree; and $e_G(U)$ for the number of edges with both endpoints in U . The graph K_n is the complete graph on n vertices. An unspecified logarithm is natural.

The main dependencies are one-way: Problem 7(a) $\rightarrow$ 7(b) $\rightarrow$ 8(a), while 8(b) is independent; Problem 9(a) is independent, the lemmas following it $\rightarrow$ 9(b), and 9(c) $\rightarrow$ 9(d)–9(e). In each of Weight Lemmas I–III the 1 : 1 case is proved first and the general case second; these lemmas are then used in Problems 18–26. Hence there are no circular references.

Contents

How to read this collection	i
A. Unvisited vertices and new edges	1
1. A Game with No Returns	1
2. Coins on a Graph	2
3. Adding Edges Until the First Odd Cycle	4
4. Adding Edges Until the Graph Is Connected	5
B. Dynamic Orientations	7
5. A Token and Edge Reversals	7
6. Two Tokens and Reversing Arrows	7
7. 2-Degenerate Graphs	9
8. Square Grids	9
9. The Bollobás–Szabó Game	10
C. Edge-Coloring and Clique Games	15
10. The First Monochromatic Triangle	15
11. Erdős’s Symmetric Game	15
12. The Biased 1 : 2 Game	18
13. Bella and Chingiz on K_n	18
14. A White Clique on an Infinite Board	21
15. Relating the 1 : 2 and 1 : 1 Games	26
D. Graphs for Games	28
16. A 2-Connected Board for Red	28
17. Companies and Geniuses	29
E. One-objective weight strategies	32
18. White Cliques	32
19. White Hamiltonian Cycles	33
20. A Family of White Targets	33
21. The Black Degree of Every Vertex	34
22. Density Inside Large Sets	34
23. Density Between Large Sets	35
F. Two-objective weight strategies	36
24. A White Clique and Black Degrees	36
25. Two Bounds in the Erdős Game	37
26. The $p : q$ Game and the Golden Ratio	39
27*. The exact Erdős conjecture	41
Addendum 2: dynamics and tournaments	42
28. Rock, Paper, Scissors on a Graph	42
29. Two Leagues	43

A. Unvisited vertices and new edges

All graphs in this section are finite and simple. In the coin problem, a strategy that guarantees a draw also counts as winning, because the task asks Petya not to lose.

1. A Game with No Returns

a) Criterion: a perfect matching

Answer. Vasya wins exactly on graphs having a perfect matching.

Suppose G has a perfect matching M . After Petya chooses the starting vertex s , Vasya moves to its mate in M . Thereafter, whenever Petya arrives at v , Vasya moves from v to its mate $M(v)$. This move is always legal. Indeed, all previously visited vertices are partitioned into fully visited pairs of M ; if $M(v)$ had already been visited, then so would v . Thus Vasya always has a reply after a Petya move. The game is finite, so Petya is the first player unable to move.

Now suppose there is no perfect matching. Let M be a matching with the largest possible number of edges, and let Petya start at an uncovered vertex s . Whenever Vasya arrives at v , Petya replies with $vM(v)$.

We check that this reply always exists. The traversed path starts at the uncovered vertex s . Petya's moves use edges of M , whereas Vasya's do not, because the matched edge would lead back to an already visited vertex. If v were also uncovered, the edges of the entire path would alternate between edges outside M and edges in M . Removing its M -edges from M and adding its other edges would produce a matching with one more edge, a contradiction. Hence $M(v)$ exists. It is unvisited: had $M(v)$ appeared earlier, v would have been visited immediately after it. Therefore Petya always has a reply, and Vasya gets stuck first.

A graph with an odd number of vertices cannot have a perfect matching, so Petya wins in that case.

b) One pass for each player

The answer is unchanged. The player possessing the strategy from part (a) follows it until the opponent passes, and immediately answers with a pass. The token and the visited set have not changed, and after the two passes the same player is to move as before them, so the original strategy resumes. If the opponent passed because no move was available, the answering pass returns the same hopeless position to that opponent, who no longer has a pass.

c) Two tokens

For every even $n \geq 4$, take the path

$$v_1 v_2 \dots v_n.$$

It has the perfect matching $v_1 v_2, v_3 v_4, \dots, v_{n-1} v_n$.

- If Petya puts the tokens at v_1 and v_n , the other $n-2$ vertices are consumed one at a time from the two ends. There are an even number of moves; Petya makes the last one, leaving Vasya without a move. Thus Petya wins.
- If Petya puts the tokens at v_1 and v_{n-1} , Vasya first moves the second token to v_{n-2} . The vertex v_n is now permanently cut off by the visited vertex v_{n-1} . The available vertices $v_2, \dots, v_{n-3}$ are then inevitably consumed from the two ends. The total number of moves is $1 + (n-4) = n-3$, which is odd. Vasya moves last and wins.

The argument also covers $n = 4$: in the second case the game ends after Vasya's single move.

d) At least two winning starts for Petya

Take a maximum matching M and an uncovered vertex s . Since s is not isolated, it has a neighbor v . The vertex v is covered by an edge $vw \in M$, for otherwise sv could simply be added to M . Replace vw by sv . The result is another maximum matching, now leaving w uncovered.

The vertices s and w are distinct. The proof of part (a) shows that Petya wins from any vertex left uncovered by some maximum matching. Thus at least the two starts s and w work.

2. Coins on a Graph

Write $w(v)$ for the number of coins at vertex v .

a) A path

Let the vertices occur in the order $v_1, \dots, v_n$, and put $a_i = w(v_i)$. Define

$$S_0 = 0, \quad S_r = a_1 - a_2 + a_3 - \dots + (-1)^{r-1} a_r.$$

If $S_n \geq 0$, Petya starts at v_1 . The subsequent route is forced, and the difference “Petya’s coins minus Vasya’s coins” is $S_n \geq 0$.

Suppose $S_n < 0$, and let r be the least index with $S_r < 0$. Then $S_{r-1} \geq 0$, and r is even: on an odd step a nonnegative term is added, so the sum cannot become negative for the first time. Petya starts at v_r . After Vasya’s first move, the token can no longer cross the visited vertex v_r , so it proceeds to one end of the path.

If Vasya goes left, the final score difference is

$$a_r - a_{r-1} + \dots - a_1 = -S_r > 0.$$

If he goes right, it is

$$a_r - a_{r+1} + a_{r+2} - \dots = S_{r-1} - S_n > 0.$$

If one direction does not exist, it need not be considered. Thus Petya can always avoid losing.

b) An arbitrary tree

Idea. We first compute the value of play in every branch, and then prove that some starting vertex has nonnegative value. Only the second step needs a separate argument.

For adjacent vertices x, y , delete xy and consider the component containing x . Let $m(x | y)$ be the best final difference

coins of the player who has just taken x – coins of the opponent,

when y is already unavailable and the opponent moves next. In a tree, all previously visited vertices are separated from x by xy , so the future position is indeed determined by $x | y$. We have the finite recursion

$$m(x | y) = w(x) - \max_{\substack{z \sim x \\ z \neq y}} m(z | x), \quad (1)$$

where the maximum over the empty set is 0. The opponent chooses a branch z in which their own advantage is $m(z | x)$, so that advantage is subtracted. Computing (1) from the leaves inward simultaneously supplies optimal strategies for both players.

If Petya starts at v , his guaranteed difference is

$$w(v) - \max_{u \sim v} m(u | v). \quad (2)$$

Call v good if (2) is nonnegative. We prove that a good vertex exists.

Suppose not. Then every vertex v has a neighbor u such that

$$m(u | v) > w(v). \quad (3)$$

Say that u threatens v . If u threatens v , then v is the unique vertex threatening u . Indeed, (1) and (3) imply

$$\max_{x \sim u, x \neq v} m(x | u) < w(u).$$

But u is not good, so some neighbor threatens it; only v can do so. Applying the same argument in the reverse direction shows that no other neighbor threatens v , either.

Thus the vertices are partitioned into adjacent pairs. Denote the mate of v by v^* , and put

$$d_v = m(v^* | v) - w(v) > 0. \quad (4)$$

Call vv^* a paired edge and every other edge a cross-edge. If uv is a cross-edge, then while computing $m(u | v)$, the unique threatening vertex u^* remains available from u . Hence

$$m(u | v) = w(u) - m(u^* | u) = -d_u. \quad (5)$$

For each vertex v , define

$$h_v = \begin{cases} 0, & \text{if } v \text{ has no cross-edge,} \\ \min\{d_u : uv \text{ is a cross-edge}\}, & \text{otherwise.} \end{cases}$$

Substituting (5) into (1) for the paired edge vv^* gives

$$m(v | v^*) = w(v) + h_v.$$

Writing this for both ends of a pair and using (4), we obtain

$$d_v + d_{v^*} = h_v + h_{v^*}. \quad (6)$$

Let $\varepsilon_v = d_v - h_v$. Equation (6) says $\varepsilon_{v^*} = -\varepsilon_v$.

Now begin a finite descent that cannot terminate. Take a leaf a of the tree. Its only edge is paired, so $h_a = 0$ and $\varepsilon_a = d_a > 0$. At a^* , however, $\varepsilon_{a^*} < 0$, so $h_{a^*} > d_{a^*}$. Hence a^* has a cross-neighbor b at which the minimum is attained:

$$d_b = h_{a^*} > d_{a^*}.$$

Because a^*b is a cross-edge, $h_b \leq d_{a^*} < d_b$, so $\varepsilon_b > 0$. Move along the paired edge to b^* ; there $\varepsilon_{b^*} < 0$, and the same argument finds a cross-neighbor c with $\varepsilon_c > 0$. Continuing gives a walk

$$a \xrightarrow{\text{paired}} a^* \xrightarrow{\text{cross}} b \xrightarrow{\text{paired}} b^* \xrightarrow{\text{cross}} c \dots$$

Its edges alternate between paired and cross-edges, so it never immediately turns back. In a tree it cannot revisit a vertex, because the segment between two visits would contain a cycle. We have constructed infinitely many distinct vertices in a finite tree, a contradiction.

Therefore a good vertex v exists. Petya starts there and, after each Vasya response, follows the strategy encoded by (1). By (2), he receives at least as many coins as Vasya.

c) A cycle

Let the cyclic sequence of weights be $a_1, \dots, a_n$.

If n is even, split the vertices into the two alternating classes. Petya starts in a class having at least as many coins in total. Whichever way Vasya moves, the rest of the cycle is traversed in that direction and Petya receives exactly his chosen class. Hence he does not lose.

Now let n be odd, with indices taken modulo n , and define

$$D_i = a_i - a_{i+1} + a_{i+2} - \dots + a_{i+n-1}.$$

This is Petya's score difference when starting at i and then moving clockwise; moving counterclockwise gives D_{i+1} . Since n is odd,

$$D_i + D_{i+1} = 2a_i \geq 0. \quad (7)$$

Thus adjacent D_i, D_{i+1} cannot both be negative. Somewhere they are both nonnegative: otherwise "negative" and "nonnegative" would have to alternate strictly around an odd cycle. Petya starts at the corresponding vertex i , and either Vasya direction gives him a nonnegative difference.

d) The assertion fails for general connected graphs

Answer. No. Here is a symmetric counterexample on seven vertices.

- Vertex z has weight 2.
- Vertices x_1, x_2, x_3 have weight 1.
- Vertices y_1, y_2, y_3 have weight 0.
- Vertex z is adjacent to every x_i .
- Vertex x_i is adjacent to y_j exactly when $i \neq j$.
- The vertices y_1, y_2, y_3 form a triangle.

We give Vasya's strategy. In every case, $\{i, j, k\} = \{1, 2, 3\}$.

1. **Petya starts at z .** Vasya goes to any x_i . Petya must choose some y_j , $j \neq i$. Vasya goes to x_k , after which Petya must go to y_i and Vasya to x_j :

$$z, x_i, y_j, x_k, y_i, x_j.$$

2. **Petya starts at x_i .** Vasya goes to z . After Petya moves to x_j , Vasya goes to y_k . Whichever of y_i, y_j Petya chooses, Vasya finishes at x_k :

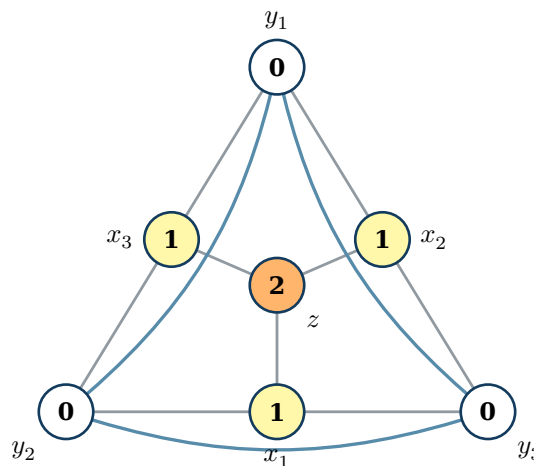
$$x_i, z, x_j, y_k, \{y_i \text{ or } y_j\}, x_k.$$

3. **Petya starts at y_i .** Vasya goes to an adjacent x_j , $j \neq i$, and let k be the third index.

- If Petya goes to z , Vasya goes to x_k , after which the moves y_j, x_i are forced.
- If Petya goes to y_k , Vasya goes to x_i . Petya can now move only to z or y_j , and in either case Vasya goes to x_k .

In every branch Vasya collects 3 coins. In the first and third cases he takes all three x -vertices; in the second, he takes z and x_k . Petya has taken, or can later take, positive-weight vertices totaling at most 2, so he cannot draw.

In the diagram, the numbers inside the vertices are coin counts. The outside arcs form the triangle $y_1y_2y_3$, keeping it visually separate from the six straight edges x_iy_j , $i \neq j$.



3. Adding Edges Until the First Odd Cycle

We first record a standard fact. A graph with no odd cycle is 2-colorable so that every edge has differently colored endpoints. In each component, choose a root and a shortest-path tree, and color a vertex by the parity of its distance from the root. If an edge uv joins equal colors, the tree path from u to v has even length, so together with uv it forms an odd cycle. Conversely, colors alternate around any cycle in a properly 2-colored graph, so every cycle has even length.

Consequently, a maximal graph still containing no odd cycle is $K_{a,b}$. Indeed, different components can always be joined safely after swapping the two colors in one component if necessary; within a connected bipartite graph, any missing edge between the color classes is safe. In $K_{a,b}$, every remaining edge lies inside one class and immediately creates a triangle. Thus the final number of safely added edges is ab .

a) Odd n

If $a + b = n$ is odd, then ab is even. Vasya always chooses a safe move while one exists. Before each Vasya move, an odd number of edges has been added, so the current graph cannot yet be the terminal $K_{a,b}$, which has an even number of edges. Vasya therefore always has a safe reply. The game is finite, so Petya is the first player forced to create an odd cycle.

Vasya wins for every odd n .

b) Even n

Answer. Petya wins when $n \equiv 2 \pmod{4}$, and Vasya wins when $n \equiv 0 \pmod{4}$.

Call a safe graph **good** if some matching covers all its nonisolated vertices. We need the following reply strategy. Suppose the graph is good before an opponent move and there are still isolated vertices, an even number of them. After the opponent's safe move, the replying player restores a good graph:

1. If two already nonisolated vertices were joined, join two isolated vertices and add this new edge to the matching.
2. If two isolated vertices x, y were joined, they are covered by xy . If old nonisolated vertices exist, join x to any one of them; this merely joins two components, and the matching still contains xy . If this is the very first reply in the empty graph, join two more isolated vertices.
3. If a nonisolated vertex was joined to an isolated vertex x , take another isolated vertex y and add xy .

Every reply is safe: it either adds an edge in a new 2-vertex component, joins different components, or attaches a new leaf. The required isolated vertices exist because, while any remain, their number before the opponent's move is even. After the reply, goodness is restored.

Let $n = 4k + 2$. Petya first adds any edge, then uses the reply strategy after every Vasya move until no isolated vertices remain. The graph now has a perfect matching. Petya thereafter chooses any safe edge. He cannot be the first player with no safe move. Otherwise the graph would be $K_{a,b}$. A perfect matching in it requires $a = b = n/2$, but then

$$ab = (n/2)^2 = (2k + 1)^2$$

is odd, whereas an even number of moves has been made before a Petya turn. Contradiction. Thus Vasya is forced to make the first losing move.

Let $n = 4k$. After Petya's first edge, the graph is good. Vasya joins two more isolated vertices and then maintains goodness by the same replies. When no isolated vertices remain, he chooses arbitrary safe moves. If Vasya were the first to have no safe move, the terminal graph would again be $K_{n/2, n/2}$, but now

$$(n/2)^2 = (2k)^2$$

is even, whereas an odd number of edges has been added before a Vasya turn. This is impossible. Hence Vasya wins.

c) Edges between two isolated vertices are forbidden

Petya wins for every even n .

Petya first adds the edge ab . He maintains symmetry under an **involution** σ , a permutation satisfying $\sigma^2 = \text{id}$: vertices a, b are fixed, and all other vertices form pairs $x, \sigma(x)$. After each Petya move the graph is invariant under σ , and the two vertices of every active pair $x, \sigma(x)$ belong to the same color class. All nonisolated vertices form one component, because after the first move the rule forbids starting a new component on two isolated vertices.

Suppose Vasya has made a safe move.

- If he attached an isolated vertex x by vx , Petya adds $\sigma(v)\sigma(x)$. The vertex $\sigma(x)$ was also isolated, so the move is legal and safe, and restores symmetry.
- If Vasya added uv between active vertices, Petya adds $\sigma(u)\sigma(v)$. This is a distinct, still missing edge. An edge fixed by σ would either be the initial edge ab , or join mates $x, \sigma(x)$; mates lie in the same color class, so such a Vasya move would already have lost. The image edge is safe because σ preserves both color classes.

Thus Petya has a safe reply after every safe Vasya move. While an isolated vertex remains, it can be joined safely to the active component, so the game cannot end before all vertices become active. Each color class then contains one of a, b and a number of whole σ -pairs. Both classes therefore have odd size. The terminal complete bipartite graph has an odd number of edges, so Petya makes the last safe move and Vasya is then forced to create an odd cycle.

4. Adding Edges Until the Graph Is Connected

Answer.

$n \pmod{4}$	winner
0	Vasya
1	Vasya
2	Petya
3	Petya

A maximal still-disconnected graph consists of two complete components $K_a \sqcup K_b$, where $a + b = n$. With three components, two can be joined safely; with a missing edge inside a component, that edge

can be added safely. The number of safe moves made in such a terminal position is

$$\binom{a}{2} + \binom{b}{2} = \binom{n}{2} - ab. \quad (8)$$

If n is odd, a, b have opposite parity, so ab is even and the parity of (8) is independent of the partition. For $n \equiv 1 \pmod{4}$, $\binom{n}{2}$ is even. Vasya always chooses a safe move: before his turn, an odd number of edges has been added, so an even-edge terminal position has not yet been reached. Hence Petya loses. For $n \equiv 3 \pmod{4}$, $\binom{n}{2}$ is odd; the same reasoning shows that Petya always has a safe move, and Vasya loses.

It remains to treat even n . In the current disconnected graph, let O be the number of odd-order components, and let R be the total number of missing internal edges:

$$R = \sum_C \left(\binom{|C|}{2} - e(C) \right).$$

The number O is even. Every safe move is of one of two types.

- **Type I:** two odd components are joined. Then O decreases by 2, while the parity of R is unchanged: the merge creates $|C||D| - 1$ new missing internal edges, an even number.
- **Type II:** any other safe move. Then O is unchanged and the parity of R changes. An internal edge decreases R by 1; a merge in which at least one component is even adds the odd number $|C||D| - 1$.

The designated player maintains after each of their moves the invariant

$$O \equiv 0 \pmod{4}, \quad R \equiv 0 \pmod{2}. \quad (9)$$

After an opponent move of Type I, $O \equiv 2 \pmod{4}$. The designated player joins two more odd components. This is safe: before the opponent's move there were at least four odd components; after the first merge, its new even component remains, so the second merge cannot make the entire graph connected. After a Type II move, R is odd and hence positive; the designated player adds any missing internal edge. Both replies are safe and restore (9).

For $n = 4k$, the empty graph satisfies the invariant and Vasya is the designated player, replying after Petya's first move. For $n = 4k + 2$, Petya first joins two isolated vertices; then O falls from $4k + 2$ to $4k$ and (9) holds, after which Petya is the replying player. The designated player never makes the graph connected; since the game is finite, the opponent loses.

Reply scheme.

opponent's move	change in $(O, R \bmod 2)$	reply
two odd components	$(-2, 0)$	join two more odd components
any other safe move	$(0, 1)$	add an internal edge

B. Dynamic Orientations

5. A Token and Edge Reversals

a) A tree

Choose a vertex r , a neighbor s , orient every tree edge toward r , and put the token at s . Petya's first move is $s \rightarrow r$.

Thereafter, before every Vasya reversal, all tree edges point toward the token. Vasya reverses an incident edge, say vu , making it the unique outgoing edge from v . Petya follows it to u . All edges now point toward u : in a tree, changing the root from v to its neighbor u changes only the direction of uv . The invariant is restored, so Petya moves forever.

b) A graph containing a cycle

We prove a stronger statement by induction on the number of vertices. In any connected graph containing a cycle, from any orientation and token position at a moment when Vasya is to reverse an edge, Vasya can leave the token with no outgoing edge after finitely many rounds.

Base case: the graph is a simple cycle. If exactly one edge leaves the token vertex, Vasya reverses it inward and traps the token immediately. If zero or two edges leave, one reversal leaves exactly one outgoing edge, which Petya must follow. Call this first move $A_1 \rightarrow A_2$. Vasya now examines the next edge A_2A_3 in the chosen direction around the cycle. If it points out of A_2 , he reverses it inward, making both edges at A_2 incoming. If it points into A_2 , he reverses it outward and Petya must move to A_3 . Vasya repeats. Either the token stops earlier or it traverses the whole cycle; upon its return to A_1 , Vasya reverses the original edge A_1A_2 inward and stops it.

Induction step. Choose a shortest cycle C . It has no chord, because a chord would split it into two cycles, one shorter. If the whole graph is not C , choose a vertex $x \notin C$ at greatest distance from C . The graph $G - x$ is connected and still contains C . Indeed, for any $y \neq x$, a shortest path from y to C cannot pass through x , or y would be farther from C than x .

Inside $G - x$, Vasya follows the induction strategy and never touches an edge incident with x . If Petya traverses an arrow $u \rightarrow x$, Vasya immediately reverses that same edge to $x \rightarrow u$. Petya must leave x back into $G - x$ on the next move, after which Vasya restarts the induction strategy from the new state.

Every such excursion to x permanently reduces the number of arrows from $G - x$ into x : Vasya reverses one of them out of x , and otherwise never touches a boundary edge. Only finitely many excursions are possible. After the last, the induction strategy in $G - x$ traps the token in finite time. If the token initially stands at x , Vasya first makes the required reversal. If exactly one arrow leaves x , he reverses it and wins at once. Otherwise he reverses any edge incident with x ; an outgoing arrow then exists, Petya's next move enters $G - x$, and the preceding strategy applies.

6. Two Tokens and Reversing Arrows

Idea. At every moment, remember only two directed routes to the target:

- P , from Petya's current token position to B ;
- Q , from Vasya's current token position to B .

The paths share no edge, though they may share vertices. Initially P is the given path from A to B , and Q is the empty path from B to itself.

Edges outside $P \cup Q$ may be forgotten. If Vasya later uses one, we see only that traversed edge and attach it to Q . The proof therefore works in an even more visual model in which Vasya may reveal new vertices and edges during play while everything outside the two colored paths disappears. This is not a new rule; it is merely a way of storing the part of the position that the strategy needs.

Here are all path updates. On Petya's move, he traverses the first edge of P ; after reversal, delete that edge from P . If P becomes empty, Petya has reached B .

Now suppose Vasya traverses $q \rightarrow r$, which then becomes $r \rightarrow q$.

1. If qr belonged to neither P nor Q , prepend $r \rightarrow q$ to Q .
2. If $q \rightarrow r$ belonged to Q , it was the first edge of Q ; delete it.

3. If $q \rightarrow r$ belonged to P , write

$$P = P_0 + (q \rightarrow r) + P_1$$

and replace the paths by

$$P \leftarrow P_0 + Q, \quad Q \leftarrow P_1. \quad (10)$$

The third update simply exchanges the two tails at q . The old paths were edge-disjoint, and the reversed edge is removed from both new routes, so the new routes remain edge-disjoint. If concatenation repeats a vertex, delete the directed cycle between its two appearances. This turns the route into a simple directed path while preserving the invariant; remove cycles in the first case in the same way.

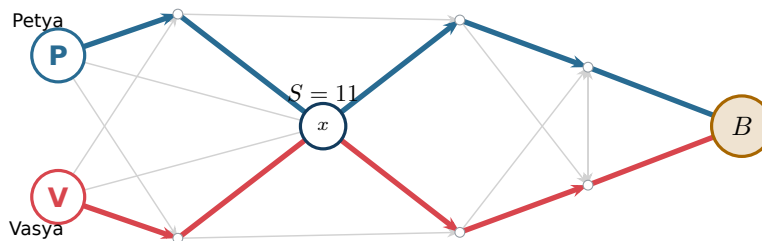
a) Vasya makes all his moves first

After each of Vasya's finitely many moves, update P, Q as above. When Vasya stops, P is still a directed path from A to B . Petya follows it in order. Each move reverses only the edge already traversed and deleted, so the remaining tail still points toward B . Petya reaches the target.

b) Alternating play

Answer. No such graph exists: Petya wins on every admissible initial graph.

The diagram shows the two remembered paths. Blue is P , red is Q , and the gray edges play no role in the strategy. The blue and red discs mark the two tokens, and the gold vertex is the target B .



In the pictured position it is Petya's turn and the path lengths are 6 and 5. Thus the displayed value is

$$|P| + |Q| + \varepsilon = 6 + 5 + 0 = 11,$$

where $\varepsilon = 0$ before a Petya move and $\varepsilon = 1$ before a Vasya move. Put

$$S = |P| + |Q| + \varepsilon. \quad (11)$$

On a Petya move, $|P|$ falls by 1 while ε changes from 0 to 1, so S is unchanged. After a Vasya move, ε changes from 1 to 0. If he moves outside $P \cup Q$, $|Q|$ grows by at most 1, so S does not increase. If he moves along Q , then $|Q|$ decreases by 1. Under the tail exchange (10), the edge qr disappears from the total length, and therefore

$$|P_{\text{new}}| + |Q_{\text{new}}| \leq |P_{\text{old}}| + |Q_{\text{old}}| - 1.$$

A Vasya pass does not change the paths. In these last three cases S strictly decreases. Thus the integer potential (11) never increases after an individual move.

Suppose the game were infinite. A nonnegative integer S cannot decrease strictly forever, so it is eventually constant. Every subsequent Vasya move must then be the only equality case: he uses an edge outside $P \cup Q$, adds exactly one edge to Q , and loses nothing when cycles are deleted. Such a move does not change P at all. But every Petya move deletes the first edge of P . After finitely many moves P is empty, meaning Petya has reached B , a contradiction.

How many Petya moves may be needed? If the initial graph has n vertices, let

$$L_0 = |P_0| \leq n - 1.$$

Measure $L = |P| + |Q|$ before Petya moves. It does not increase during a full round. While $L = k$ and does not decrease, Vasya must be in the equality case and hence leaves P unchanged, while Petya shortens P by one. Within at most k Petya moves, either L decreases or Petya wins. Thus the total number of Petya moves is at most

$$L_0 + (L_0 - 1) + \dots + 1 \leq \frac{n(n-1)}{2}.$$

The diagram illustrates a position, not the upper bound itself; Vasya can in fact delay the win for many rounds in a suitable 12-vertex example.

7. 2-Degenerate Graphs

a)

Answer. The two conditions are equivalent.

Recall that a graph is 2-degenerate if every nonempty subgraph has a vertex of degree at most 2.

Suppose the graph is 2-degenerate. Repeatedly remove a vertex of degree at most 2 from the remaining graph, obtaining an order $v_1, v_2, \dots, v_n$, where v_i is removed before v_{i+1} . Direct every edge from the later removed endpoint toward the earlier removed one.

Only edges joining v_i to vertices still present when v_i was removed point into v_i , so its indegree is at most 2. Along every arrow the index decreases; hence no directed cycle exists.

Conversely, suppose an acyclic orientation with indegree at most 2 is given. Restrict it to any nonempty subgraph. A finite acyclic digraph has a **sink**, a vertex with no outgoing arrow: otherwise, repeatedly following an outgoing arrow would eventually repeat a vertex and form a cycle. Every edge of the subgraph at a sink points into it, so its degree is at most 2. Thus the graph is 2-degenerate.

b)

Choose the orientation D from part (a). At each vertex with exactly two incoming edges in D , pair those two edges. The pairs are disjoint because every edge enters a unique vertex.

The key observation is this: if, in the actual game, both edges of every such pair are directed in the same way relative to their common vertex (both in or both out), then there is no directed cycle. Indeed, restrict D to the edges of a hypothetical cycle. Since D is acyclic, it has a sink on this cycle; the two cycle edges at that vertex form one distinguished pair. But in an actual directed cycle one enters that vertex and the other leaves, a contradiction.

Cycle-Breaker's strategy is now a pairing strategy.

- If $|E(G)|$ is even and Cycle-Maker starts, the number of edges outside the distinguished pairs is also even. Pair them arbitrarily. After every Cycle-Maker move, Cycle-Breaker takes the other edge of that pair. For a distinguished pair, she directs the second edge so that the two agree relative to their common vertex; in an arbitrary pair, its direction is irrelevant.
- If $|E(G)|$ is odd and Cycle-Breaker starts, at least one edge is outside the distinguished pairs. She directs one such edge arbitrarily on her first move, pairs the remaining unpaired edges, and then replies as above.

We must also check that Cycle-Maker cannot complete a cycle before her reply. Before each Cycle-Maker move, every distinguished pair is either untouched or complete and coherently directed relative to its center. If the new arrow closed a cycle, take a sink of D on that cycle. Its two incident cycle edges form a distinguished pair. The pair could not have been completed earlier, because its directions would contradict the cycle; but it could not be untouched, because both its edges now lie on the cycle. Contradiction.

Thus a directed cycle appears after neither player's move, and Cycle-Breaker wins.



In the left diagram both reference edges enter x ; on a directed cycle, one of them would have to enter and the other leave.

8. Square Grids

a)

Answer. Cycle-Breaker wins on the infinite square grid for either first player.

First suppose Cycle-Maker starts. As a reference orientation D , direct all horizontal edges to the right and all vertical edges upward. Exactly two reference edges enter each vertex, from the left and from below; pair them. After a Cycle-Maker move, Cycle-Breaker takes the other edge of its pair and directs it in the same way relative to the common vertex. The full argument of Problem

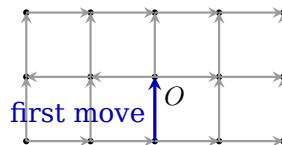
7(b) excludes a cycle both after her reply and immediately after a Cycle-Maker move. The infinite graph causes no difficulty: every candidate winning cycle is finite, and the argument concerns only its edges.

Now suppose Cycle-Breaker starts. Her first move directs $(0, -1)(0, 0)$ upward. Use the modified reference orientation:

- every vertical edge points upward;
- on the row $y = 0$, horizontal edges point away from the origin;
- on every other row, horizontal edges point to the right.

It is acyclic. Along a vertical arrow the y -coordinate strictly increases, so a directed cycle could contain no vertical edge; each individual row is an oriented path and has no cycle.

Exactly one reference edge enters the origin, namely the first edge already oriented. Exactly two enter every other vertex; pair them and use the reply strategy from Problem 7(b). The only unpaired reference edge has already been played, so no directed cycle can arise.



The first move is blue; at every other vertex the two incoming arrows form a pair.

b)

The number of edges is

$$m(n-1) + n(m-1) = 2mn - m - n,$$

so

$$|E(P_m \square P_n)| \equiv m + n \pmod{2}.$$

Here is an independent pairing strategy. Mentally orient every horizontal edge to the right and every vertical edge upward. At each vertex having a left neighbor and a lower neighbor, pair the two edges arriving from them. Leave the other edges temporarily unpaired.

If the total number of edges is even and Cycle-Maker starts, an even number of edges remains and can be paired arbitrarily. If the total is odd and Cycle-Breaker starts, the number remaining is odd and positive. She directs one arbitrarily on her first move, then pairs the rest. Thereafter she always takes the mate of Cycle-Maker's edge. In a structural pair, she makes the two edges agree relative to their common vertex; in an arbitrary pair, direction does not matter.

Why is a cycle impossible? In any grid cycle, choose a vertex with maximum sum of coordinates. Both incident cycle edges go left or down and hence form one structural pair. Yet in a directed cycle one must enter and the other leave, whereas the strategy directs them alike relative to the vertex. A cycle cannot arise immediately after Cycle-Maker's move either: before it, every structural pair is untouched or complete. In the former case both edges could not yet lie on the new cycle; in the latter their directions already contradict a cycle.

Thus Cycle-Breaker wins. Since the edge parity equals the parity of $m + n$, these are precisely the two required cases. For $m = n = 1$, there are no edges and the assertion is automatic.

9. The Bollobás-Szabó Game

A sink was defined in Problem 7; a **source** is a vertex with no incoming arrow.

a)

In every component of a 4-regular graph, construct a closed trail using each edge exactly once. Walk along unused edges. One cannot get stuck away from the starting vertex: before an arrival at a vertex, an even number of its incident edges has been used, and just after the arrival an odd number has been used, so not all four. If edges remain after returning, start another such trail at a vertex of the old trail incident with an unused edge and splice it in. This produces an Euler tour.

Orient each tour in its traversal direction. Exactly two edges enter every vertex; pair them at that vertex. These pairs partition all edges.

Cycle-Maker first takes one edge of any pair, directs it out of the center, and sets its mate aside. Thereafter:

- if Cycle-Breaker takes an edge of an untouched pair, Cycle-Maker takes its mate and directs it out of the center;
- if Cycle-Breaker takes the set-aside edge, Cycle-Maker opens a new pair, directs one edge out of its center, and sets the other aside;
- if no new pair remains, Cycle-Breaker's move on the last set-aside edge ends the orientation and no reply is needed.

At least one arrow leaves the center of every pair, so the final orientation has no sink. Every finite digraph without a sink has a directed cycle: follow outgoing arrows until a vertex repeats. Hence Cycle-Maker wins.

We need a more general pairing assignment for the remaining parts.

Lemma 1 (assignment to admissible places). Suppose each of finitely many objects may occupy certain places. If every set of q objects has at least q admissible places in total, then all objects can be assigned to distinct admissible places.

Proof. Place as many objects as possible. If an object p is unplaced, mark it, then mark all its admissible places, then all objects already occupying marked places, and repeat these two steps. If a marked free place appears, shift the objects successively along the resulting chain and place p , contradicting maximality.

Thus no marked place is free. All marked places are occupied, while there is one more marked object: their occupants plus the original p . Moreover, every place admissible for a marked object is marked. This gives fewer admissible places than objects, contrary to the hypothesis. $\square$

Lemma 2 (pairs in a minimal dense core). A **minimal dense core** is a graph H with $v(H) \geq 3$, $e(H) = 2v(H) - 2$, such that every proper subgraph F with $v(F) \geq 3$ has at most $2v(F) - 3$ edges. Let H be such a core and let the first marked edge be directed $a \rightarrow b$.

1. Every edge can be assigned to one endpoint so that no edge is assigned to a , exactly two edges are assigned to every other vertex, and ab is assigned to b .
2. Suppose there is another edge $c \rightarrow d$, distinct from ab , with $a \neq c$. The assignment may additionally be required to assign cd to c .

The two edges assigned to one vertex will be called its *pair*.

Proof. Create two free places at every vertex except a . An edge may be placed only at one of its endpoints. In the first case, place ab in advance at one place of b . In the second, also place cd at one place of c . If $b = c$, these two edges occupy the two different places of that vertex.

We verify Lemma 1 for the remaining edges. Let Q be any set of them, and let X be the set of all their endpoints other than a . Write $e_H(S)$ for the number of edges with both endpoints in S . Every edge of Q is counted by $e_H(X \cup \{a\})$.

If this is a proper vertex set and $|X| \geq 2$, minimality gives

$$e_H(X \cup \{a\}) \leq 2(|X| + 1) - 3 = 2|X| - 1. \quad (1)$$

If $b \in X$, the edge ab is counted here but no longer belongs to Q . Therefore in all cases

$$|Q| \leq 2|X| - 1 - \mathbf{1}_{b \in X}. \quad (2)$$

In the first case, the vertices of X have $2|X| - \mathbf{1}_{b \in X}$ free places, and (2) is even one stronger than needed.

In the second case, the number of remaining places is

$$2|X| - \mathbf{1}_{b \in X} - \mathbf{1}_{c \in X}.$$

If $c \in X$, this equals the right-hand side of (2); if $c \notin X$, (2) is again one below the number of free places. Notice that we do not need to count cd inside $X \cup \{a\}$: its endpoint d might not belong to this set, and the strict unit in (1) pays for the lost place.

If $X \cup \{a\} = V(H)$, the total number of remaining edges equals the total number of remaining places: respectively $2v(H) - 3$ and $2v(H) - 4$, so there are certainly enough places for Q . If $X = \emptyset$, then Q

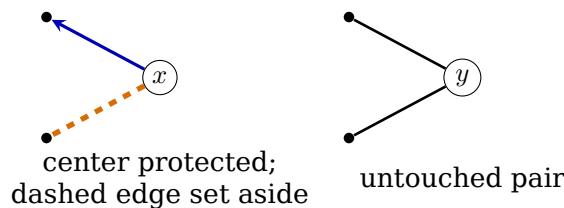
is empty. If $X = \{x\}$, it can contain only the single edge ax . In the first case there are $2 - \mathbf{1}_{x=b}$ free places, and when $x = b$ the edge $ax = ab$ was already placed. In the second case there are $2 - \mathbf{1}_{x=b} - \mathbf{1}_{x=c}$ places; zero can occur only for $x = b = c$, when again $ax = ab$ was placed already. All small cases are covered. Lemma 1 now assigns every remaining edge. $\square$

The pair-and-set-aside mechanism. If the two edges of a pair are assigned to x , Cycle-Maker can *protect* x from becoming a sink by directing at least one of them out of x .

Suppose several pairs remain untouched and perhaps one edge is *set aside*; the center to which it is assigned is already protected.

- If Cycle-Breaker takes an edge of an untouched pair, Cycle-Maker takes the other and directs it out of the pair's center.
- If Cycle-Breaker takes the set-aside edge, Cycle-Maker opens any untouched pair, directs one edge out of its center, and makes the other the new set-aside edge.
- If no untouched pair remains, the move on the last set-aside edge simply completes the orientation.

Thus Cycle-Maker gradually protects every center.



b)

Let Cycle-Breaker's first move be $a \rightarrow b$, and apply Lemma 2(1). Vertex a is already not a sink, and ab lies in the pair assigned to b . Cycle-Maker immediately takes the other edge of this pair and directs it out of b .

After every later Cycle-Breaker move, Cycle-Maker takes the mate in the same pair and directs it out of the center. Every vertex other than a therefore also gets an outgoing arrow. The final orientation has no sinks. The rules of this part require play to continue even if a cycle appears earlier; the strategy remains defined throughout.

c)

Suppose Cycle-Maker first directs $a \rightarrow b$, and Cycle-Breaker replies $c \rightarrow d$. First assume $a \neq c$. Apply Lemma 2(2), assigning ab to b and cd to c . The first edge protects a , and the second protects c .

If $b \neq c$, Cycle-Maker takes the other edge of b 's pair and directs it out of b , while setting aside the unused mate of c 's pair. If $b = c$, the first two edges form one pair and already protect its center; Cycle-Maker opens any untouched pair and sets its second edge aside. He then uses the mechanism above. Unless a directed cycle appeared earlier, the final orientation has no sinks.

It remains to consider $a = c$. Since the two moves used distinct edges, $b \neq d$. Reverse every arrow mentally. The first arrows become $b \rightarrow a$ and $d \rightarrow a$, with distinct initial vertices. The case already proved guarantees no sinks after reversal, which means no sources in the original orientation. A directed cycle remains directed when every arrow is reversed. Thus Cycle-Maker guarantees one of the required alternatives.

d)

Among subgraphs of G having at least $2v - 2$ edges and at least three vertices, choose an inclusion-minimal one H . Then

$$e(H) = 2v(H) - 2,$$

since otherwise one edge could be deleted. Every proper subgraph $F \subsetneq H$ with $v(F) \geq 3$ has at most $2v(F) - 3$ edges, or it could replace H . Hence H is a minimal dense core.

Cycle-Maker plays only on H and starts with any arrow $a \rightarrow b$. A Cycle-Breaker move outside H is simply a pass in the game on H .

- If Cycle-Breaker's first reply lies in H , use part (c). After Cycle-Maker's next move inside H , untouched pairs and one set-aside edge remain.

- If the first reply lies outside H , use Lemma 2(1) for $a \rightarrow b$. Cycle-Maker takes the mate in b 's pair and directs it out of b . Only untouched pairs remain inside H .

Handle later passes as follows. If there is no set-aside edge and Cycle-Breaker moves outside H , Cycle-Maker opens a new pair and sets its mate aside. If a set-aside edge exists and Cycle-Breaker again moves outside, Cycle-Maker simply orients that edge, leaving none set aside. A move in an untouched pair is answered on its mate; a move on the set-aside edge opens the next pair. If no appropriate edge remains, every edge of H is oriented.

Thus, regardless of moves outside H , Cycle-Maker obtains on H an orientation with no sinks or no sources, unless a directed cycle appeared earlier. In the “no sources” branch, all replies are reversed: toward, not away from, each pair center. A finite orientation with no sinks or with no sources contains a directed cycle. Hence Cycle-Maker wins.

The dependencies are one-way:

$$\text{Lemmas 1-2} \rightarrow 9(b), 9(c) \rightarrow 9(d).$$

Part 9(a) is not used in the proof of 9(d).

e)

A 4-regular graph on N vertices has

$$2e = 4N, \quad e = 2N.$$

If Cycle-Maker starts, part (a), or part (d), gives a win. If Cycle-Breaker starts, ignore her first oriented edge. The remaining unoriented graph has $2N - 1 \geq 2N - 2$ edges, and Cycle-Maker now moves first among them. Part (d) produces a directed cycle using only these remaining edges.

Now consider a cellular embedding on a closed orientable surface of genus $g \geq 1$. “Cellular” means that every component of the complement is a disk; these components are the faces. If there are f faces, every face has four sides and every edge borders two faces, so

$$4f = 2e, \quad f = \frac{e}{2}.$$

For this cell decomposition,

$$N - e + f = 2 - 2g.$$

Here is a short proof rather than an appeal to the formula as a black box. The quantity $V - E + F$ is unchanged by two refinements. Subdividing an edge adds one vertex and one edge. Drawing a simple arc inside a face to split it adds one edge and one face. Any two finite cell decompositions of the same surface have a common refinement: perturb their edges slightly, mark all intersections, and subdivide the resulting regions by arcs if needed. We may therefore compute $V - E + F$ on the standard genus- g polygon with $4g$ sides identified in $2g$ pairs. After identification it has one vertex, $2g$ edges, and one face, so $V - E + F = 1 - 2g + 1 = 2 - 2g$.

Substituting $f = e/2$ yields

$$e = 2N + 4g - 4.$$

For $g \geq 1$, this is at least $2N$. If Cycle-Maker starts, apply part (d). If Cycle-Breaker starts, at least $2N - 1$ edges remain after her move, and part (d) applies again. Cycle-Maker wins in either move order.

f)

Problem 8(a) says that Cycle-Breaker wins on the infinite square grid in both move orders. We construct finite graphs on which Cycle-Maker wins in both.

For $R \geq 0$, put

$$L = 2R + 3, \quad G_R = C_L \square C_L.$$

Equivalently,

$$V(G_R) = (\mathbb{Z}/L\mathbb{Z})^2,$$

with two vertices adjacent when they differ by ± 1 in exactly one coordinate. Take $v_R = (0, 0)$. The parameters are

$$|V(G_R)| = L^2 = (2R + 3)^2, \quad |E(G_R)| = 2L^2 = 2(2R + 3)^2.$$

The graph is simple and 4-regular because $L \geq 3$. Part (e) gives Cycle-Maker a win in either move order.

It remains to check the rooted ball. Every vertex of the radius- R ball has a unique representative $(x, y) \in \mathbb{Z}^2$ with

$$|x| + |y| \leq R.$$

Two such representatives cannot coincide modulo L , because the absolute difference in each coordinate is at most $2R < L$. Nor is there an extra edge across the torus seam: an absolute coordinate difference is at most $2R$, whereas false adjacency would require $L - 1 = 2R + 2$. Thus adjacency inside the ball is exactly the same as in $\mathbb{Z}^2$.

The rooted radius- R balls are therefore isomorphic, while the winner in each move order is opposite.

C. Edge-Coloring and Clique Games

10. The First Monochromatic Triangle

Answer. Blue wins for every $n \geq 5$.

The strategy uses at most five vertices, chosen partly after Red's first reply. Blue starts with P_1P_2 .

Case 1: Red's first edge shares an endpoint with P_1P_2 . Relabel so that it is P_2P_3 , and choose fresh vertices P_4, P_5 . Blue colors P_2P_4 , threatening P_1P_4 and the blue triangle $P_1P_2P_4$. Red must take P_1P_4 ; with only two red edges, Red cannot yet win.

Blue now takes P_2P_5 . On the next move, either of

$$P_1P_5, \quad P_4P_5$$

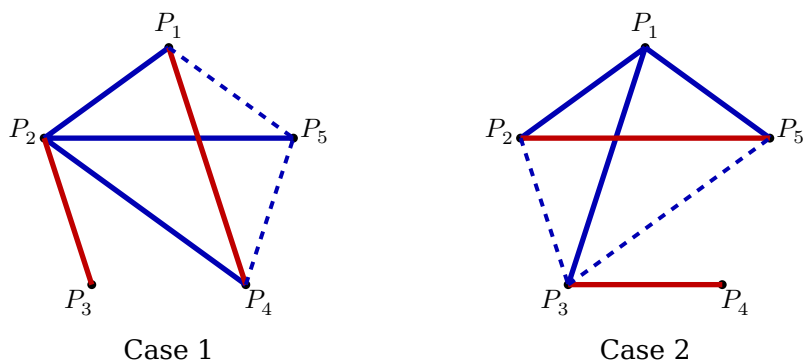
wins, completing $P_1P_2P_5$ or $P_2P_4P_5$, respectively.

Case 2: Red's first edge is disjoint from P_1P_2 . Call it P_3P_4 and choose a fifth vertex P_5 . Blue colors P_1P_5 , threatening P_2P_5 , which Red must take. Blue then takes P_1P_3 , creating the two threats

$$P_2P_3 \quad \text{and} \quad P_3P_5.$$

They complete $P_1P_2P_3$ and $P_1P_3P_5$, respectively.

In either case, before Red's third move the two red edges are disjoint: P_2P_3, P_1P_4 or P_3P_4, P_2P_5 . One third edge cannot complete a red triangle, because any two edges of a triangle share an endpoint. Red can stop only one of Blue's two threats, so Blue realizes the other and wins. If Red ever fails to stop an earlier single threat, Blue wins sooner. The hypothesis $n \geq 5$ is needed exactly to choose five distinct vertices.



Solid blue and red edges show the position after Blue's third move; the two dashed edges are the simultaneous blue threats.

A one-vertex set also counts as a clique. Write ω_R, ω_B for the largest red and blue clique sizes, and $\alpha(G)$ for the independence number of G .

11. Erdős's Symmetric Game

We first record a device used several times below.

Imaginary-edge device. If a player can win even after several free edges have been gifted to the opponent, then the player can certainly win the real game. Regard the gifted edges as already claimed by the opponent. If the opponent later actually takes one, count that move as a pass and make any forced extra move elsewhere. An extra edge of one's own color cannot shrink one's clique or enlarge the opponent's. The same convention handles a move made after the part of the board under consideration is already full.

a)

Suppose, for a contradiction, that Blue has a strategy guaranteeing

$$\omega_B > \omega_R$$

against every Red play. Red first colors any edge e , then pretends to be the second player and follows Blue's strategy with the color names exchanged. In the imagined game, actual blue edges

play the role of first-player edges, and all red edges except e play the role of second-player edges. The strategy would therefore give

$$\omega(R - e) > \omega(B).$$

The extra red edge e can only help. The same strategy would promise a win to both players, a contradiction. Thus Red can guarantee $\omega_R \geq \omega_B$.

b)

The game is finite, so backward induction gives one of two alternatives: either Blue can ensure $\omega_R \leq \omega_B + 1$, or Red can ensure $\omega_R \geq \omega_B + 2$. We show that the second is impossible.

Let S be such a hypothetical Red strategy, and let f be its first move. Consider Red's actual first edge e . Blue tries to play S with the colors exchanged.

- If $e \neq f$, Blue colors f and then literally follows S . Applied to the blue graph and the red graph without e , S would imply

$$\omega_B \geq \omega(R - e) + 2 \geq \omega_R + 1.$$

The last inequality uses the fact that deleting one edge lowers clique number by at most one. This already contradicts Red's promised $\omega_R \geq \omega_B + 2$.

- If $e = f$, Blue makes any free move g , then ignores it. In the imagined play of S , the first blue edge is e ; actual Red moves from the second onward are the opponent's moves, and actual Blue moves from the second onward are the strategy's replies. Thus

$$\omega((B - g) + e) \geq \omega(R - e) + 2. \quad (*)$$

Adding one edge increases clique number by at most one: from a clique using that edge, delete one endpoint. Hence the left side of $(*)$ is at most $\omega(B) + 1$, while the right side is at least $\omega(R) + 1$. Therefore $\omega_B \geq \omega_R$.

If at most one free edge remains after e , the required bound is immediate, since Red has only one edge. Thus in the second branch we may assume the imagined game continues after g . This detail matters only for the arbitrary board in part (c); on K_n , $n \geq 3$, there are enough edges anyway.

In both branches the promised Red strategy and the Blue strategy constructed from it cannot both fulfill their guarantees. This contradiction proves the claim.

c)

Neither proof above used completeness of the board; they used only finiteness and the fact that one edge changes clique number by at most one. Hence both arguments apply unchanged to every finite graph G . Under optimal play, Red guarantees a difference at least zero and Blue at most one, so the difference is 0 or 1.

d)

Let Red's first move on K_{n+1} be uv . Strengthen Red by gifting all other edges incident with v . The remaining n vertices form a fresh K_n , on which Blue moves first. Blue uses the assumed winning first-player strategy. A Red move outside this board is a pass in the imagined game; after the K_n board is full, Blue colors any remaining free edge.

If the largest red and blue cliques inside this K_n have sizes r, b , then $b \geq r + 1$. In the whole graph, a red clique may additionally use the universal red vertex v , while a blue clique cannot. Therefore

$$\omega_R = r + 1 \leq b = \omega_B.$$

Blue does not lose and hence wins by the rules of the problem.

e)

Suppose Red starts with xy , and let $H = K_{n+2} - \{x, y\} \cong K_n$. Blue's first move begins the assumed winning first-player strategy on H . Thereafter:

1. a red move inside H is answered inside H according to that strategy;
2. a move xz , $z \in H$, is answered by yz , and conversely.

The reply in the second rule is always free, because the pair $\{xz, yz\}$ could previously have been touched only together.

If an internal strategic reply was already used as an extra blue move, or no edge remains inside H , Blue takes any free edge. The imaginary-edge device shows that this accounting does not change the strategy's promise on H .

Let the clique numbers inside H be r, b , so $b \geq r + 1$. A red clique of size at least three cannot contain both x, y : for every third vertex, one of the two required edges is blue. Hence

$$\omega_R \leq r + 1 \leq b \leq \omega_B.$$

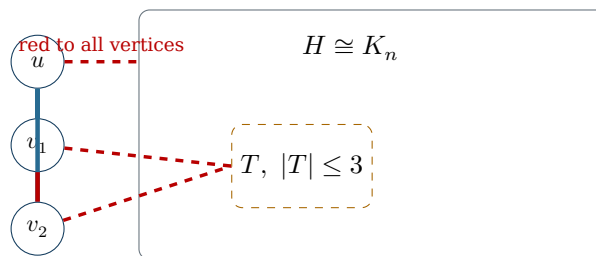
f)

Fix a winning first-player strategy S on K_n . We first describe a target position from which Blue can finish easily.

Target position. Partition the vertices into $H \cong K_n$ and three special vertices u, v_1, v_2 , with:

- inside H , Blue is playing first according to S and currently has one more edge than Red, unless H is already complete;
- every edge from u to H is red, while uv_1, uv_2 are blue;
- v_1v_2 is red, and every common red neighbor of v_1, v_2 in H lies in a set T with $|T| \leq 3$.

Solid edges in the diagram are already colored; dashed edges are imaginary gifts to Red. The rectangle is H , and the small region inside is T .



Why the target position wins. Blue answers a red move inside H according to S , and pairs v_1w with v_2w . A move on an imaginary gifted edge counts as a pass. Thus, if the final clique numbers inside H are r, b , then $b \geq r + 1$. A red clique using at most one special vertex has size at most $r + 1$. A clique using two special vertices cannot contain u together with either v_1 or v_2 , so it consists of v_1, v_2 and at most their three common red neighbors. Hence

$$\omega_R \leq \max\{r + 1, 5\}. \quad (1)$$

We prove the rough Ramsey estimate needed here. Every red-blue coloring of K_{70} contains a monochromatic K_5 . More generally, induction on $a + b$ shows that among $\binom{a+b-2}{a-1}$ vertices there is a red K_a or a blue K_b . At a chosen vertex, either at least $\binom{a+b-3}{a-2}$ incident edges have one color, or at least $\binom{a+b-3}{a-1}$ have the other; apply induction to the corresponding neighbors. For $a = b = 5$, this gives $\binom{8}{4} = 70$.

Since $n > 100$, the graph H has a red or blue K_5 . In the latter case $b \geq 5$; in the former, $r \geq 5$ and hence $b \geq r + 1 \geq 6$. Thus $b \geq 5$, and (1) yields $\omega_B \geq \omega_R$.

Reaching the target position. Let x be an endpoint of Red's first edge. Blue first takes xy , with fresh endpoint y . After Red's second move, choose z so that no red edge joins it to x or y , and so that every already drawn red edge avoiding x, y has endpoint z . This is always possible: there is at most one such red edge; use a suitable endpoint if it exists, otherwise take a fresh vertex. No colored edge then lies inside

$$H = K_{n+3} - \{x, y, z\}.$$

Blue begins S on H and answers every red move in H until Red first takes an edge e outside H ; e may be xz or yz . Blue takes a free edge from $\{xz, yz\}$ distinct from e . Before e , both were free, so this is possible. The one of x, y now joined in blue to both other special vertices is named u ; the remaining two are v_1, v_2 .

If Red has just taken the last edge of H , Blue has no internal strategic reply. He immediately takes either xz or yz and assigns the names; the target position is already reached with H complete.

Up to this moment Red has drawn at most three edges outside H . Thus the union of existing red neighbors of v_1, v_2 inside H has size at most three; include it in T . Gift Red every missing edge from

u to H , both edges from each vertex of T to v_1, v_2 , and $v_1 v_2$. None is blue. This is exactly the target position and completes the proof.

g)*

Statement. Prove or disprove that there is an integer N such that for every $n \geq N$, if Red can force a strict win on K_n , then Blue has a winning strategy on K_{n+4} .

Open problem. In accordance with the project statement, no solution is given here.

h)*

Statement. Prove or disprove that Blue wins under optimal play for every $n \geq 3$.

Open problem. In accordance with the project statement, no solution is given here.

12. The Biased 1 : 2 Game

a)

At the end, every edge of K_n outside the white graph W is black. A vertex set is therefore a black clique exactly when no two of its vertices are joined by a white edge, that is, exactly when it is independent in W . Thus

$$\omega(\text{black graph}) = \alpha(W).$$

b)

After the first white edge uv , call the other vertices x, y . Chingiz immediately colors xy and any other free edge black. Every later white edge shares an endpoint with uv , because the unique disjoint edge is already black.

On K_4 , Bella makes exactly two moves. Her white graph is therefore two adjacent edges plus an isolated vertex. It has $\omega(W) = 2$, while three vertices form an independent set, so $\alpha(W) = 3$. Part (a) gives a black clique of size $3 > 2$, and Chingiz wins.

13. Bella and Chingiz on K_n

a) Even n , four black edges

Idea. Chingiz partitions the vertices into black pairs. Almost every block of four edges between two pairs becomes entirely black except for one white edge. The exceptional blocks form a forest, and every oriented forest has a source.

Initially all vertices are *free*. During play, Chingiz creates disjoint pairs $\{x, \bar{x}\}$ with black edge $x\bar{x}$. Call the four edges between two pairs their *block*. Some blocks are marked; the marked blocks form a graph F whose vertices are the pairs.

Reply to a white edge xy as follows.

1. If x, y are free, choose two further free vertices $\bar{x}, \bar{y}$, color

$$x\bar{x}, \quad y\bar{y}, \quad \bar{x}\bar{y}$$

black, create pairs $X = \{x, \bar{x}\}$, $Y = \{y, \bar{y}\}$, and mark the block XY .

2. If x belongs to $X = \{x, \bar{x}\}$ and y is free, choose a free $\bar{y}$, color

$$y\bar{y}, \quad x\bar{y}, \quad \bar{x}\bar{y}$$

black, create $Y = \{y, \bar{y}\}$, mark XY , and provisionally orient $X \rightarrow Y$.

3. If both endpoints already belong to pairs, color every still-free edge of their block black. There are at most three if the block is unmarked and at most two if it is marked.

The first two rules use at most three edges. With the fourth edge, Chingiz removes the only possible obstruction: if exactly two free vertices remain after the new pairs are formed, color the edge between them black and declare them a pair. Thus before every Bella move there are either no free vertices or at least four, and Rule 1 is always legal. Any unused quota is spent arbitrarily.

The graph F is a forest. Rule 1 adds an edge between two new vertices, and Rule 2 attaches one new vertex to an old one. A pair created by the fourth edge is isolated in F .

Orient the remaining edges of F after play. In a Rule-1 block, the first white edge is xy , while $\bar{x}\bar{y}$ is black. Of the other two edges $x\bar{y}, \bar{x}y$, at most one can become white: after that move, Rule 3 immediately colors the other. If $x\bar{y}$ is black, orient $X \rightarrow Y$; if it is white, orient $Y \rightarrow X$. We obtain the property

$$\boxed{X \rightarrow Y, p \in X, q \in Y, pq \text{ white} \implies p\bar{q} \text{ black.}} \quad (*)$$

For a Rule-2 block this already holds under the orientation $X \rightarrow Y$, since both $x\bar{y}$ and $\bar{x}y$ are black.

If free vertices remain at the end, no white edge is incident with them; otherwise they would have been paired by Rule 1 or 2. All their incident edges are therefore black. Pair these vertices arbitrarily and add them to F as isolated vertices.

Let C be a largest white clique, $|C| = s$. Bella made a move, so $s \geq 2$. It contains at most one vertex from each pair. Let $\bar{C}$ be the set of opposite vertices.

The set $\bar{C}$ is a black clique. In an unmarked block, all edges except the first white one become black. In a marked block, the edge opposite the first white edge is made black immediately, and the edge opposite a possible second white edge is made black by Rule 3.

In the oriented subforest of F on the pairs represented in C , choose a source pair X , and let $c \in C \cap X$. For every other $q \in C$, the edge $c\bar{q}$ is black. If their pairs are adjacent in F , this follows from (*) and the fact that X is a source. If they are not adjacent, the block is unmarked, so everything except the white edge cq is black. Also $c\bar{c}$ is a black pair edge. Hence

$$\bar{C} \cup \{c\}$$

is a black clique of size $s + 1$, and Chingiz wins.

b) Even n , three black edges

For parts (b) and (c), use a more economical strategy. If a rule prescribes fewer than three new black edges because some are already black, Chingiz uses the remaining quota arbitrarily. A block is **early** if its first white edge appears before both pairs have been created. Maintain:

- every opposite-pair edge is black;
- the edge opposite every white edge between pairs is black;
- the early blocks form a forest on the pairs.

Let $n = 2m$. Vertices initially have no names. While at least four unnamed vertices remain, reply as follows.

1. After a white edge uv between two unnamed vertices, choose unnamed $\bar{u}, \bar{v}$ and color

$$u\bar{u}, \quad \bar{u}\bar{v}, \quad v\bar{v}$$

black.

2. If u already has a mate and v is unnamed, choose its mate $\bar{v}$ and color

$$\bar{u}\bar{v}, \quad v\bar{v}$$

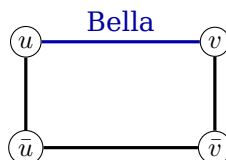
black.

3. If both endpoints have mates, after white uv color every still-free edge among

$$u\bar{v}, \quad \bar{u}v, \quad \bar{u}\bar{v}$$

black.

The first case is illustrated below: white uv creates two opposite pairs, and the opposite edge $\bar{u}\bar{v}$ immediately becomes black.



If exactly two unnamed vertices x, y remain after a round, use

Bella's edge	Chingiz's edges	(2)
xv	$x\bar{v}, yv, y\bar{v}$	
yv	$xv, x\bar{v}, y\bar{v}$	
xy	no special reply is needed.	

Rule 3 still handles moves between already paired vertices.

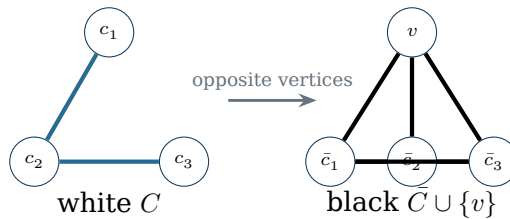
We prove the win and the invariants together. Early blocks form a forest: an edge between unnamed vertices creates an isolated forest edge on two new pairs, while an edge between a named and an unnamed vertex attaches a new pair to an old one. An edge between two existing pairs is not early.

A non-early block has at most one white edge. In an early block, the first white edge has a black opposite; after the next white move, Rule 3 colors all remaining edges. Hence an early block has at most two white edges, and if there are two, they share an endpoint.

Let C be a white clique of s paired vertices. It contains at most one vertex per pair. The opposite set $\bar{C}$ is a black clique because the edge opposite every edge of C is black. A white edge between C and $\bar{C}$ can occur only in an early block between two represented pairs, and there is at most one in each such block. The forest induced by these s pairs has at most $s - 1$ edges. Hence at most $s - 1$ edges between C and $\bar{C}$ are white. Some $v \in C$ has no white neighbor in $\bar{C}$, and therefore

$$\bar{C} \cup \{v\}$$

is a black clique of size $s + 1$.



If a largest white clique contains x but not y , write it as $C \cup \{x\}$. By (2), y is joined in black to all of $\bar{C}$ and to the vertex v just found, so $\bar{C} \cup \{v, y\}$ is one vertex larger. The case with y is symmetric. The white edge xy belongs to no white triangle, since for every third vertex (2) makes one of the two links black. If the white clique is just $\{x, y\}$, then before these two leftover vertices appeared there was already a white edge in the paired part; the forest argument for $s = 2$ gives a black triangle. Thus the largest black clique is always strictly larger.

c) Odd n , three black edges

Chingiz begins exactly as in part (b), creating opposite pairs. Stop this phase at the first of two events:

- three unnamed vertices remain and Bella first touches one of them;
- five unnamed vertices remain and Bella takes an edge between two of them.

Let L be the set of paired vertices.

Case 1. The remaining vertices are a, b, c , and Bella's first new edge is av_0 , $v_0 \in L$. Chingiz colors the triangle abc black. Thereafter, after a white edge dv with $d \in \{a, b, c\}$, $v \in L$, he colors every still-free edge among

$$a\bar{v}, b\bar{v}, c\bar{v}$$

black; inside L , use Rule 3 from part (b). The only exception is the initial white av_0 : if it occurs among prescribed edges after $d\bar{v}_0$, simply skip it and color the others.

A white clique contains at most one of a, b, c . If it is $C \cup \{d\}$, $C \subset L$, $|C| = s$, then

$$(\bar{C} \setminus \{v_0, \bar{v}_0\}) \cup \{a, b, c\}$$

is a black clique of size at least $s + 2$. A white clique contained in L is handled by part (b).

Case 2. The remaining vertices are a, b, c , and the first new white edge is ab . Chingiz colors ac, bc black. Continue with

$$\begin{array}{c|c} av & bv, b\bar{v}, c\bar{v} \\ bv & av, a\bar{v}, c\bar{v} \\ cv & a\bar{v}, b\bar{v}, c\bar{v}. \end{array} \quad (3)$$

The old rule still applies inside L . If a white clique is $C \cup \{a\}$, then $\bar{C} \cup \{b, c\}$ is a black clique one vertex larger. The case of b is symmetric; for c , use $\bar{C} \cup \{a, c\}$. No white clique can contain a, b and a vertex of L , by the first row of (3). The edge ab itself is harmless. For $n = 5$, Case 3 occurs immediately. For $n \geq 7$, before the last three vertices appeared, some ordinary pairing move produced a white edge inside L ; the forest argument with $s = 2$ gives a black K_3 in L .

Case 3. Five vertices remain. Call the endpoints of the first white edge x, y , and the others a, b, c . Chingiz colors the triangle abc black. Thereafter:

1. after one of av, bv, cv , color $a\bar{v}, b\bar{v}, c\bar{v}$;
2. after xv or yv , color the other three edges between $\{x, y\}$ and $\{v, \bar{v}\}$;
3. if Bella colors ax , color ay, by, cy ; if she later takes bx or cx , color the remaining one. The other five variants follow by renaming;
4. inside L , use the old rule.

A white clique with at most one of the five special vertices is handled as in part (b), with extra slack from the black triangle for a, b, c . The edge xy lies in no white triangle. If a white clique contains two special vertices, then, up to renaming, it is

$$C \cup \{a, x\}.$$

The set

$$\bar{C} \cup \{b, c, y\}$$

is a black clique one vertex larger. This exhausts all cases, so Chingiz wins for every odd $n > 3$.

All prescribed replies are legal. Rule 3 uses only free edges. In an early block, after the second white move every other free edge becomes black. For an edge between a special vertex and v , if a required edge to $\bar{v}$ had earlier become white, the reply to that earlier move would already have made the current edge to v black. Similarly, under the ax -rule, an earlier white edge among ay, by, cy would by the symmetric response have made ax black. Thus no table ever asks Chingiz to recolor a white edge; already black prescribed edges do not count as new, and unused quota is spent on arbitrary free edges.

d)*

Statement. Prove or disprove that for $k = 2$, Chingiz has a winning strategy for every even $n > 3$.

Open problem. In accordance with the project statement, no solution is given here.

14. A White Clique on an Infinite Board

Throughout this problem, a Bella move means one white edge. Thus $T_b(r)$ counts white moves, not the total number of colored edges.

a) How Bella builds a clique

First construct a finite auxiliary tree with levels

$$0, 1, \dots, r-1.$$

Level 0 consists of one vertex, the **root**. Every vertex on levels $0, \dots, r-2$ has exactly $b+1$ children. The vertices on the last level are the **leaves**. We call x an ancestor of y if the path from the root to y passes through x .

The vertices of this tree are merely positions in an auxiliary diagram. Bella gradually assigns actual vertices of K_∞ to some of them. Call such a tree vertex **occupied**, and call the corresponding graph vertex its **representative**.

Bella maintains the following property:

if two occupied tree vertices lie on one downward path from the root, then their representatives are joined by a white edge.

Initially Bella occupies the root with an arbitrary representative. She then repeats the following attempt.

1. Bella chooses a new actual vertex u having no black edge to any representative chosen so far. Such a vertex exists: only finitely many representatives and black edges have appeared, whereas K_∞ has infinitely many vertices.
2. Bella starts at the root of the auxiliary tree. Suppose she is currently at an occupied vertex x . She colors the edge from u to the representative of x white.
3. Below x begin $b + 1$ branches, one through each child. Chingiz's reply consists of only b black edges, so he can join u by a black edge to an occupied representative in at most b of these branches. At least one branch remains untouched, and Bella chooses it.
4. If the first child in the chosen branch is already occupied, Bella moves to it and repeats steps 2–3. Otherwise she assigns u as its representative and ends the attempt.

Why is the property preserved? During the attempt, Bella joined u by white edges to every occupied vertex on her downward path. Those vertices are exactly the occupied ancestors of the new position assigned to u .

Black replies cannot stop the descent: at each level there are $b + 1$ continuations and only b black reply edges. Moreover, every black edge played at an earlier level of the same attempt points into a branch already discarded.

If Bella reaches an occupied vertex on level $r - 2$, her next move creates a white K_r . Indeed, the path from the root to this vertex contains $r - 1$ occupied vertices. Their representatives form a white clique, and during this attempt the new vertex u has been joined in white to every one of them.

Finally, the process cannot continue forever. Every unsuccessful attempt occupies a new vertex of the finite auxiliary tree. Sooner or later Bella reaches the last level and wins.

This also gives an immediate move bound. To occupy a vertex on level j , Bella draws j white edges, and that level has $(b + 1)^j$ vertices. Even if she must occupy every vertex on levels $1, \dots, r - 2$ and then one leaf, she uses at most

$$T_b(r) \leq \sum_{j=0}^{r-2} j(b+1)^j + (r-1). \quad (4)$$

In particular, $T_b(r)$ is finite.

b) The exact answer for a triangle

We prove both inequalities.

Bella needs at most $b + 3$ moves. She chooses a vertex v and draws $b + 2$ white edges from it:

$$vx_1, vx_2, \dots, vx_{b+2}.$$

Each new x_i is chosen so that all its edges to previously named vertices are still uncolored. This is possible for the same reason as in part (a): only finitely many edges have been colored.

After the move vx_{b+2} , all $b + 1$ edges from x_{b+2} to $x_1, \dots, x_{b+1}$ are free. Any one of them completes a white triangle with v . Chingiz can color at most b of these edges black. Bella takes the remaining one on her next move. Hence $T_b(3) \leq b + 3$.

Bella cannot win before move $b + 3$. Call a free edge **dangerous** if coloring it white would immediately complete a white triangle. After each of Bella's first $b + 1$ moves, Chingiz colors every newly dangerous edge black and uses any remaining moves arbitrarily.

He has enough moves. Suppose Bella has just drawn her t -th white edge uv . Every new dangerous edge must form a triangle together with uv and one earlier white edge. There are $t - 1$ earlier white edges, and each of them creates at most one new dangerous edge. Thus there are at most $t - 1 \leq b$ new dangerous edges.

After Chingiz's reply to Bella's $(b + 1)$ -st move, no free dangerous edge remains. Therefore Bella still cannot complete a triangle on her $(b + 2)$ -nd move. Consequently,

$$T_b(3) = b + 3.$$

Preparation for parts (c) and (d): a weight lemma

Consider first a general finite game. Let X be a finite set of game elements; in our application these are the edges of a complete graph. Let $\mathcal{F}$ be a family of targets, each target $A \in \mathcal{F}$ being a subset

of X . Bella wins by claiming every element of at least one target. On each turn she claims one free element and Chingiz claims b free elements.

Put $\lambda = b + 1$. For a target A containing no Chingiz element, let $u(A)$ be its number of free elements and assign it weight

$$\lambda^{-u(A)}.$$

If Chingiz has claimed an element of a target, that target is no longer available to Bella and its weight is zero. Let Φ be the total weight of all available targets.

Weight lemma. If before play begins

$$\sum_{A \in \mathcal{F}} \lambda^{-|A|} < \frac{1}{\lambda}, \quad (5)$$

then Chingiz can prevent Bella from completing any target.

Proof. Bella's first move multiplies the weight of each affected target by at most λ . Hence immediately after that move $\Phi < 1$.

For each of his b choices, Chingiz uses the following greedy rule. For every free element, add the weights of all available targets containing it, and choose an element for which this sum is largest.

Consider Bella's next element x . After all b Chingiz choices, suppose the total weight of the available targets containing x is L . At each of Chingiz's choices, x was still free. Therefore the element chosen by Chingiz had incident target weight at least L , and its choice destroyed all those targets. His b choices reduced the total weight by at least bL .

Bella then takes x . Every surviving target containing x has its weight multiplied by λ , increasing the total weight by

$$(\lambda - 1)L = bL.$$

Thus Φ , measured immediately after Bella's moves, never increases and always remains below 1. A completed target would have weight 1, so Bella can never complete one. This proves the lemma.

Why the finite lemma applies to K_∞ . Suppose Chingiz only needs to defend against Bella's first t moves. During that time at most $(b+1)t$ edges are colored, with at most $2(b+1)t$ endpoints. Consider the finite graph

$$K_N, \quad N = 2(b+1)t.$$

As actual vertices of K_∞ appear, assign them distinct names from K_N . If the finite strategy asks for an edge using a name not yet assigned, assign that name to a fresh actual vertex. There are enough names, since at most N endpoints occur in t rounds. Hence any defending strategy on K_N transfers to the first t moves on K_∞ .

c) Bounds for $T_1(r)$

Lower bound. Set

$$t = \lfloor 2^{r/2} \rfloor - 1, \quad N = 4t < 2^{r/2+2}.$$

In K_N , every r -vertex clique gives a target consisting of its $\binom{r}{2}$ edges. There are $\binom{N}{r}$ such targets. For $b = 1$ we have $\lambda = 2$, so (5) becomes

$$\binom{N}{r} 2^{-\binom{r}{2}} < \frac{1}{2},$$

equivalently

$$\binom{N}{r} < 2^{\binom{r}{2}-1}. \quad (6)$$

We verify (6). Since

$$\binom{N}{r} < \frac{N^r}{r!} < \frac{2^{r^2/2+2r}}{r!},$$

it is enough to prove

$$r! > 2^{5r/2+1}. \quad (7)$$

For $r = 17$, this is true:

$$17! = 355687428096000 > 2^{43.5}.$$

When r increases to $r+1$, the left-hand side is multiplied by $r+1 \geq 18$, while the right-hand side is multiplied only by $2^{5/2} < 6$. Thus (7), and hence (6), holds for every $r \geq 17$.

The weight lemma lets Chingiz defend through Bella's first t moves. Hence

$$T_1(r) \geq t + 1 = \lfloor 2^{r/2} \rfloor.$$

Upper bound. Put $b = 1$ in (4). The identity

$$\sum_{j=0}^{r-2} j2^j = (r-3)2^{r-1} + 2$$

follows immediately by induction. Therefore

$$T_1(r) \leq (r-3)2^{r-1} + r + 1.$$

d) Bounds for $T_2(r)$

Lower bound. Now set

$$t = \lfloor 3^{r/2} \rfloor - 1, \quad N = 6t < 6 \cdot 3^{r/2}.$$

Here $b = 2$ and $\lambda = 3$. As in part (c), the weight lemma applies if

$$\binom{N}{r} < 3^{\binom{r}{2}-1}. \quad (8)$$

We have

$$\binom{N}{r} < \frac{N^r}{r!} < \frac{6^r 3^{r^2/2}}{r!}.$$

Thus it suffices for (8) that

$$r! > 6^r 3^{r/2+1}. \quad (9)$$

At $r = 30$, compare directly:

$$\begin{aligned} 30! &= 265252859812191058636308480000000, \\ 6^{30} 3^{16} &= 9516507342594806772904803434496. \end{aligned}$$

The first number is larger. As r increases, the left-hand side of (9) is multiplied by $r+1 \geq 31$, while the right-hand side is multiplied by only $6\sqrt{3} < 12$. Thus (9) holds for every $r \geq 30$, and the weight lemma gives

$$T_2(r) \geq \lfloor 3^{r/2} \rfloor.$$

Upper bound. Putting $b = 2$ in (4) gives the slightly stronger estimate

$$T_2(r) \leq \sum_{j=0}^{r-2} j3^j + r - 1.$$

In particular,

$$T_2(r) \leq \sum_{j=0}^{r-2} j3^j + r + 1.$$

Moreover,

$$\sum_{j=0}^{r-2} j3^j \leq (r-2) \sum_{j=0}^{r-2} 3^j = \frac{r-2}{2} (3^{r-1} - 1),$$

and for $r \geq 2$ the last expression plus $r+1$ is less than $r3^{r-1}$. Therefore

$$T_2(r) < r3^{r-1}.$$

e) Beck's bound

Here $b = 1$. Bella wins on her first move when $r = 2$, so assume $r \geq 3$. We need one auxiliary game.

Half-selection lemma. Let F be a graph on an even number of vertices. Two players alternately choose one previously unchosen vertex, with First moving first. Then First can ensure that the vertices chosen by First span at most $e(F)/4$ edges of F .

Proof. Track only edges neither of whose endpoints has yet been chosen by Second; call them surviving edges. If a surviving edge has s unchosen endpoints, give it weight 2^{-s} . Initially every edge has weight $1/4$, so the total weight is $e(F)/4$.

First chooses a vertex with minimum incident weight among surviving edges. That move doubles the weight of each such incident edge, so it increases the total weight by exactly this minimum. Second then chooses a vertex and destroys all surviving edges incident with it. Before First's move, the incident weight at Second's chosen vertex was at least the minimum, and during First's move it could only increase. Thus Second destroys at least as much weight as First has just added.

After every pair of moves, total weight does not increase. At the end, the surviving edges are precisely those with both endpoints chosen by First, and each has weight 1. There are therefore at most $e(F)/4$ such edges.

Now put

$$n = r + 2$$

and choose a set V_0 of 2^n vertices in the infinite graph. Bella will construct nested sets

$$V_0 \supset V_1 \supset V_2 \supset \dots$$

and vertices $u_1, u_2, \dots$ such that u_i is joined in white to every vertex of V_i .

The first step. Bella chooses $u_1 \in V_0$ and draws white edges from it until she has

$$s_1 = 2^{n-1}$$

white neighbors; these form V_1 . This is possible because u_1 has $2^n - 1 = 2s_1 - 1$ edges to $V_0 \setminus \{u_1\}$, and between two Bella moves Chingiz can take at most one of them.

Let e_i be the number of black edges with both endpoints in V_i , let $s_i = |V_i|$, and let δ_i be the minimum black degree within V_i . During the first step Chingiz makes at most s_1 moves, so

$$e_1 \leq s_1, \quad \delta_1 \leq \frac{2e_1}{s_1} \leq 2. \quad (10)$$

The last inequality follows because the sum of black degrees in V_1 is $2e_1$.

Constructing V_i from V_{i-1} . Choose a vertex $u_i \in V_{i-1}$ having exactly δ_{i-1} black neighbors within V_{i-1} . Delete u_i and all those black neighbors, and take a largest even subset A of the remaining vertices.

Every edge from u_i to A is free. None is black by the definition of A . None is white either: at earlier stages the only white edges were those from u_j to V_j , and $u_j \notin V_j$. Thus no white edge has yet been drawn inside any V_j .

Let F be the graph of existing black edges within A , and imagine playing the half-selection lemma on its vertices. Whenever First chooses x , Bella colors $u_i x$ white. If Chingiz replies with $u_i y$, where $y \in A$ is still unchosen in the imagined game, declare that Second chose y . In every other case, assign any still unchosen vertex of A to Second.

Bella's prescribed spoke is always free. Indeed, if Chingiz had colored the spoke to some vertex earlier, that vertex was assigned to Second at once and can never be chosen by First. Let V_i be the vertices chosen by First. Then

$$s_i = \left\lfloor \frac{s_{i-1} - 1 - \delta_{i-1}}{2} \right\rfloor. \quad (11)$$

By the lemma they span at most one quarter of the old black edges. During this stage Chingiz adds at most one new black edge for each of Bella's s_i moves. Hence

$$e_i \leq \frac{e_{i-1}}{4} + s_i. \quad (12)$$

The sets remain large enough. We prove by induction that for $1 \leq i \leq n - 4$,

$$2^{n-i} - 7 \leq s_i \leq 2^{n-i}, \quad \delta_i \leq 5. \quad (13)$$

For $i = 1$, this follows from the definition of s_1 and (10).

Assume the claim for $i - 1$, and set

$$x = 2^{n-i}.$$

Then $s_{i-1} \geq 2x - 7$ and $\delta_{i-1} \leq 5$, so (11) gives

$$s_i \geq \frac{(2x - 7) - 5 - 2}{2} = x - 7.$$

The extra subtraction of 1 accounts for rounding down. The upper bound $s_i \leq x$ is immediate from (11).

Iterating (12), we obtain

$$e_i \leq s_i + \frac{s_{i-1}}{4} + \frac{s_{i-2}}{4^2} + \dots.$$

Since $s_{i-k} \leq 2^k x$,

$$e_i < s_i + x,$$

because

$$\sum_{k \geq 1} \frac{2^k}{4^k} = 1.$$

In this range $x \geq 16$ and $s_i \geq x - 7$. Thus the average black degree within V_i is less than

$$\frac{2(s_i + x)}{s_i} \leq 2 + \frac{2x}{x-7} \leq 2 + \frac{32}{9} < 6.$$

Some vertex therefore has at most five black neighbors, so $\delta_i \leq 5$. This completes the induction.

Finishing the construction. At $i = n - 4$, (13) gives

$$|V_{n-4}| \geq 16 - 7 = 9, \quad \delta_{n-4} \leq 5.$$

Bella chooses a vertex u_{n-3} of minimum black degree in V_{n-4} . After deleting it and its at most five black neighbors, at least one vertex u_{n-2} remains, and the edge $u_{n-3}u_{n-2}$ is free. Bella colors this edge white.

The vertices

$$u_1, u_2, \dots, u_{n-2}$$

now form a white clique. Indeed, if $j < k$, then $u_k \in V_j$, while u_j was joined in white to every vertex of V_j ; the last pair is the edge Bella has just colored. The clique has $n - 2 = r$ vertices.

Constructing V_i used s_i moves, followed by one final move. The upper bound in (13) yields

$$\sum_{i=1}^{n-4} s_i + 1 \leq \sum_{i=1}^{n-4} 2^{n-i} + 1 < 2^n = 2^{r+2}.$$

Together with the case $r = 2$, this proves for every $r \geq 2$ that

$$T_1(r) < 2^{r+2}.$$

f)*

Statement. For large r , narrow the gap between the best known bounds

$$2^{r/2} \leq T_1(r) \leq 2r^7 2^{2r/3}.$$

Comment. The upper bound is due to Gebauer and is substantially stronger than Beck's bound, but its known proof is too technical for this project.

Open problem. In accordance with the project statement, no solution is given here.

15. Relating the 1 : 2 and 1 : 1 Games

Choose a vertex z . In the first phase Bella repeatedly takes a free edge at z , continuing until none remains. Of the $M - 1 \geq 6N$ star edges, Chingiz can take at most one per round, so Bella obtains a set S of at least $3N$ white neighbors of z . Let $|S| = s$. Chingiz has made at most s moves by then, so the black graph induced by S has at most s edges.

We use the elementary fact that a graph on s vertices with e edges has an independent set of size at least

$$\frac{s^2}{2e + s}. \tag{12}$$

Take a random ordering of the vertices and select a vertex if it occurs before all its neighbors. The selected vertices are independent, and a degree- d vertex is selected with probability $1/(d+1)$. The expected number selected is $\sum_v 1/(d(v)+1)$, which is at least (12), since for positive $a_1, \dots, a_s$,

$$\left(\sum a_i\right) \left(\sum \frac{1}{a_i}\right) \geq s^2;$$

after expansion this is just the sum of $a_i/a_j + a_j/a_i \geq 2$. Some ordering therefore attains (12).

Here $e \leq s$, so S has an independent set X of size at least $s/3 \geq N$. Keep exactly N of its vertices.

In the second phase, Bella runs on X her winning 1 : 2 strategy. The real Chingiz takes only one edge per round. If it lies in X and is still free in the imagined game, count it as the first of two imagined black edges. If it lies outside X , or was already imagined black in an earlier round, count it as a pass. Then mentally color arbitrary free edges inside X black until the imagined round has two black moves, or the board is full. Every real black edge inside X is accounted for, so the white move prescribed by the original strategy is always really free.

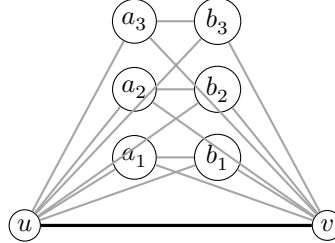
Bella obtains a white K_r inside X . All its vertices are joined in white to z , so together with z they form a white K_{r+1} .

D. Graphs for Games

16. A 2-Connected Board for Red

For $t \geq 2$, construct H_t from t copies of K_4 sharing exactly one common edge uv . In copy i , call the other vertices a_i, b_i and call the copy the **page** C_i .

three pages are shown; their number is arbitrary



Take

$$t \geq \max \left\{ 2, \left\lceil \frac{M-2}{2} \right\rceil \right\}.$$

Then H_t has $2 + 2t \geq M$ vertices.

It is 2-connected. After deleting u , the vertex v joins all remaining vertices; after deleting v , u does. After deleting any a_i or b_i , the common edge uv still connects every page. Thus deleting one vertex always leaves a connected graph.

Red first takes uv . Blue's first edge lies in some page C_j . Red chooses a different untouched page C_1 and takes the opposite edge a_1b_1 . Rename the page of Blue's first edge as C_2 .

Use the following local replies in other pages.

- Page C_2 already contains one blue edge e . If $e = a_2b_2$, Red pairs

$$\{ua_2, ub_2\}, \quad \{va_2, vb_2\}.$$

If, for instance, $e = ua_2$, use

$$\{a_2b_2, ub_2\}, \quad \{va_2, vb_2\};$$

the other spokes are symmetric. A blue edge in a pair is answered by its mate.

- Suppose Blue first enters an untouched page C_i . If he takes a_ib_i , Red takes ua_i , pairs va_i, vb_i , and answers one with the other if needed. If the first blue edge is a spoke, say ua_i , Red immediately takes a_ib_i . No blue triangle in that page is then possible. All variants follow by renaming.

Red gives priority to every required local reply. If none is required, he works in the target page C_1 . Pair its four spokes as opposite edges

$$P = \{ua_1, vb_1\}, \quad Q = \{ub_1, va_1\}.$$

Red takes a free edge from whichever pair he has not yet represented. If both are represented, he may use a free move such as a_ib_i in an untouched page, which immediately makes that page safe from a blue triangle.

Red must eventually obtain an edge from both P and Q . If Blue first enters one of these pairs, no local reply is due, and Red immediately takes its mate. If Blue avoids C_1 , the finitely many other pages are exhausted and he must eventually enter it. Any two edges, one from P and one from Q , share an endpoint; the third edge of the corresponding red triangle is one of uv, a_1b_1 .

It remains to check Blue's defense. Every triangle of H_t lies in one page. In C_1 , the opposite edges uv, a_1b_1 are red. In C_2 , every possible blue triangle requires both edges of one displayed pair. In a new page, after the first blue spoke, a_ib_i is red; after the first blue a_ib_i , the triangle through u is blocked by red ua_i , and the triangle through v requires both edges of the pair $\{va_i, vb_i\}$. Triangles through uv are impossible because uv is red.

Thus $\omega_R \geq 3$ and $\omega_B \leq 2$, so Red wins strictly.

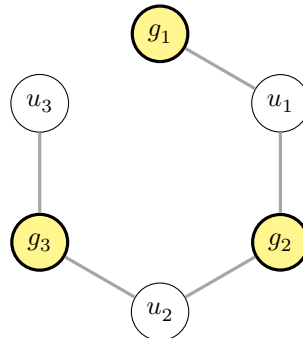
17. Companies and Geniuses

a) A hexagon

Take the cycle

$$g_1 - u_1 - g_2 - u_2 - g_3 - u_3 - g_1$$

and designate g_1, g_2, g_3 as geniuses.



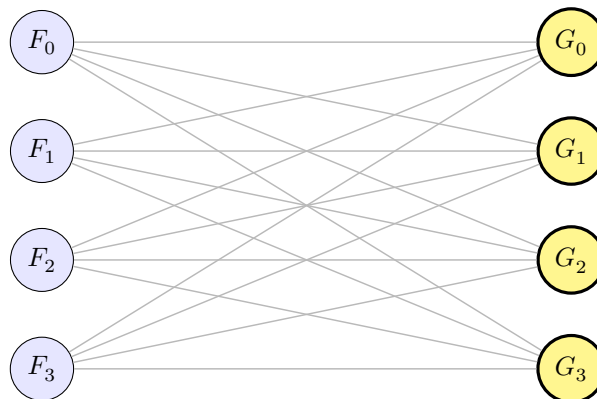
If Petya first hires a genius, say g_1 , Vasya hires the opposite vertex u_2 . If Petya next takes u_1 , Vasya takes g_2 ; if Petya takes u_3 , Vasya takes g_3 . Play around the cycle is then forced, and the second remaining genius also goes to Vasya.

If Petya starts at an ordinary vertex, say u_1 , Vasya takes the adjacent genius g_1 . Petya is forced to take g_2 , Vasya takes u_3 , Petya is forced to take u_2 , and Vasya takes g_3 . In either case Vasya gets two geniuses.

b) Four famous programmers

Besides geniuses G_0, G_1, G_2, G_3 , take four famous programmers F_0, F_1, F_2, F_3 . For each i, j , join F_i to G_j by a separate path of length k_{ij} ; the internal vertices of all sixteen paths are distinct, and there are no other edges. The lengths are

	G_0	G_1	G_2	G_3
F_0	13	12	11	10
F_1	12	11	10	13
F_2	11	10	13	12
F_3	10	13	12	11

(13)


each segment is a separate path;
its length is listed in the table

The construction has $8 + \sum_{i,j} (k_{ij} - 1) = 176$ vertices. For comparison, the general construction in part (c) has $\binom{15}{3} = 455$ vertices when $r = 4$.

Define a company's distance to a free programmer as the least number of its future hires needed to reach that programmer. Vasya chooses his first worker after seeing Petya's as follows.

1. If Petya chooses a famous programmer, Vasya chooses another famous programmer closer to three geniuses. For example, after F_0 , take F_1 : Petya's distances to G_0, G_1, G_2 are 13, 12, 11, while Vasya's are 12, 11, 10. Other cases are cyclic relabelings.

2. If Petya chooses a genius, call it G_3 . Vasya takes any famous programmer. His distance to each of G_0, G_1, G_2 is at most 13, while Petya must pass through a famous programmer and hence needs at least $10 + 10 = 20$ moves.
3. If Petya starts at an internal vertex of the path $F_i G_3$, Vasya takes F_i . That path is now blocked for Petya through F_i , so to reach any of G_0, G_1, G_2 , Petya must first reach G_3 and then traverse two more paths, needing at least $1 + 10 + 10 = 21$ moves. Vasya again needs at most 13.

Call the selected three geniuses G_0, G_1, G_2 . From his famous programmer, Vasya approaches each along the direct path in the construction. Petya cannot enter any of these three paths from the other end without first hiring that genius.

Vasya's rule is: if Petya advances along a path toward one of the three selected geniuses, Vasya advances toward the same genius; if Petya's move approaches none of them, Vasya advances toward the one currently farthest from him.

After every Vasya move, his distance to each of the three geniuses is strictly smaller than Petya's. At an internal path vertex this is immediate: Petya can decrease at most one distance by one, and Vasya decreases that same distance. Several distances can drop together only when Petya reaches a famous vertex or G_3 . Before such a branching point, however, Petya has traversed a path of length at least 10, while Vasya distributed those moves among his most distant targets. In Cases 2 and 3 there is also an initial margin of at least $20 - 13 = 7$. Direct substitution of the four path lengths 10, 11, 12, 13 confirms that all inequalities remain strict at every branching point.

If Petya could hire a selected genius, just before his move his distance to it would be 1, while Vasya's would be less than 1; that genius would already work for Vasya, a contradiction. Vasya therefore hires all three.

c) The general construction

Put $L = r(r - 1)$. The programmers are all vectors

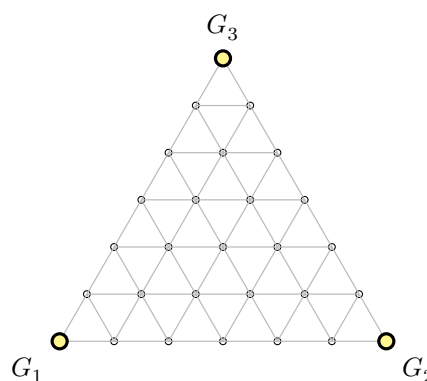
$$x = (x_1, \dots, x_r), \quad x_i \in \mathbb{Z}_{\geq 0}, \quad x_1 + \dots + x_r = L. \quad (14)$$

Two programmers know each other if one vector is obtained from the other by increasing one coordinate by 1 and decreasing another by 1. Designate as geniuses

$$G_i = (0, \dots, 0, \underset{i\text{-th coordinate}}{L}, 0, \dots, 0), \quad 1 \leq i \leq r.$$

The graph is finite and has exactly r geniuses.

For $r = 3$, it is a triangular grid with the three corner vertices as geniuses.



Suppose Petya first hires $A = (A_1, \dots, A_r)$. Some coordinate, say A_k , is at least $L/r = r - 1$. Vasya first hires

$$B_i = A_i + 1 \quad (i \neq k), \quad B_k = A_k - (r - 1). \quad (15)$$

All coordinates are nonnegative and still sum to L , so this programmer exists.

For each company and each $i \neq k$, track the largest i -th coordinate among its employees. Write m_i for Petya's maximum and M_i for Vasya's. Immediately after (15),

$$M_i > m_i \quad (i \neq k). \quad (16)$$

Vasya maintains all these inequalities.

On one move Petya hires a neighbor of a previous employee. Exactly one coordinate of the new vector increases, and only by one. Thus after a Petya move at most one inequality in (16) can become equality, and none can reverse.

If $m_i = M_i = d$, Vasya has an employee S with $S_i = d$. We have $d < L$: for $d = L$, the unique programmer is the genius G_i , who could not belong to both companies. Hence $S_j > 0$ for some $j \neq i$. Vasya hires the neighbor

$$S' = S + e_i - e_j, \quad (17)$$

where e_i is the i -th unit vector. The move is legal, and S' is unhired because its i -th coordinate is $d + 1$, above both previous maxima. Thus $M_i > m_i$ is restored.

If no equality occurs, choose any $i \neq k$ with $M_i < L$ and use the same move (17) to increase M_i . The new neighbor is again unhired because its new coordinate exceeds both companies' maxima. Every Vasya move therefore increases $\sum_{i \neq k} M_i$ by one until all these maxima reach L . Vasya cannot get stuck earlier; if Petya stops hiring, Vasya simply continues.

The equality $M_i = L$ means that Vasya has hired the unique vector with that coordinate, namely G_i . He eventually hires every genius except possibly G_k , at least $r - 1$ geniuses.

E. One-objective weight strategies

Weight Lemma 1: white targets

First, the 1 : 1 game. Call a target alive if it contains no black edge. If it still has r free edges, give it weight 2^{-r} ; give a dead target weight 0. The danger of a free edge is the sum of the weights of the alive targets containing it.

Chingiz colors an edge of maximum danger D black. The potential, meaning the total weight of all targets, decreases by D . The danger of every remaining edge is now at most D . When Bella colors one of them white, the weights of all targets containing it double, so the potential increases by at most D . Thus the potential does not increase during a round. At the end, every completely white target has weight 1, and hence

$$N_{\mathcal{F}} \leq \left\lfloor \sum_{F \in \mathcal{F}} 2^{-|F|} \right\rfloor. \quad (1a)$$

Now the general $p : q$ game. In each round Chingiz colors p free edges black, and then Bella colors at most q free edges white. Let $\mathcal{F}$ be a family of targets, where each $F \in \mathcal{F}$ is a nonempty set of edges. A target remains alive until one of its edges becomes black. Put

$$\lambda = (p + 1)^{1/q}.$$

If an alive target F has $r(F)$ free edges, give it weight $w(F) = \lambda^{-r(F)}$; a dead target has weight zero. The danger of a free edge e is the total weight of the alive targets containing e .

Chingiz's strategy. He chooses each of his p edges with maximum current danger, recalculating after every black edge of the same move.

Let the dangers of the chosen edges be $d_1 \geq d_2 \geq \dots \geq d_p$, and let the maximum danger of a remaining free edge after Chingiz's move be d . Then $d \leq d_p$, and the total weight has fallen by $d_1 + \dots + d_p \geq pd$.

One white edge multiplies the weight of every alive target containing it by λ . Before Bella's j -th white edge, the danger of any edge is at most $\lambda^{j-1}d$. Hence the total increase during Bella's move is at most

$$(\lambda - 1)d(1 + \lambda + \dots + \lambda^{q-1}) = (\lambda^q - 1)d = pd.$$

Thus a full round does not increase the total weight. If Chingiz's last move is incomplete, he takes all remaining edges and Bella does not move again; if Bella's last move is incomplete, the increase is only smaller. At the end, an alive target is completely white and has weight 1. Therefore

$$N_{\mathcal{F}} \leq \left\lfloor \sum_{F \in \mathcal{F}} (p + 1)^{-|F|/q} \right\rfloor. \quad (1)$$

For $p = q = 1$, formula (1) reduces to (1a).

18. White Cliques

For each k -vertex set, take as a target all $\binom{k}{2}$ edges among its vertices. Part (a) follows from the short 1 : 1 lemma, (1a), and part (b) from the general formula (1).

For part (c), put

$$k = \lceil 2q \log_{p+1} n \rceil + 2.$$

Then $k - 1 > 2q \log_{p+1} n$, so

$$\binom{n}{k} (p + 1)^{-\binom{k}{2}/q} \leq n^k (p + 1)^{-k(k-1)/(2q)} < 1.$$

If $k > n$, the required estimate is trivial. If $k \leq n$, part (b) says that no white k -clique exists. In both cases,

$$\omega(W) \leq k - 1 = \lceil 2q \log_{p+1} n \rceil + 1.$$

The 1 : 1 case is obtained by taking $p = q = 1$.

19. White Hamiltonian Cycles

Let $h(W)$ denote the number of Hamiltonian cycles in the white graph, where changing the starting vertex or reversing the direction does not create a new cycle.

The graph K_n has $(n-1)!/2$ Hamiltonian cycles: fix a starting vertex, order the remaining vertices, and divide by two for the two traversal directions. Every cycle contains n edges. Part (a) follows from (1a), while the general formula (1) gives in part (b)

$$h(W) \leq \left\lfloor \frac{(n-1)!}{2(p+1)^{n/q}} \right\rfloor.$$

For $p = q = 1$, the denominator is 2^{n+1} . If the expression inside the floor is less than one, no white Hamiltonian cycle exists. This is part (c).

20. A Family of White Targets

This is exactly Weight Lemma I: its targets F are the members of $\mathcal{F}$. Part (a) is (1a), and part (b) is the general formula (1).

Weight Lemma II: an almost proportional share of black edges

First, the 1 : 1 game. Choose $c > 1$, and then choose $s > 1$ sufficiently close to 1 that

$$s - 1 \leq 1 - s^{-c}.$$

Give target A weight $a_A s^{w_A - cb_A}$. Define the danger of an edge as

$$(1 - s^{-c}) \sum_{A \ni e} a_A s^{w_A - cb_A}.$$

A black edge of maximum danger D decreases the potential by D , while the next white edge increases it by at most $(s-1)D/(1-s^{-c}) \leq D$. Hence the potential does not increase. Since c may be chosen arbitrarily close to 1, the share $1/(c+1)$ may be made arbitrarily close to $1/2$ from below.

Now the general $p : q$ game. Associate a set of edges with every target A , and let w_A, b_A be the numbers of its already white and already black edges. First choose

$$c > \frac{q}{p}.$$

Then choose $s > 1$ sufficiently close to 1 that

$$s^q - 1 \leq p(1 - s^{-c}). \quad (2)$$

Such an s exists: as $s \downarrow 1$, after division by $s-1$, the left- and right-hand sides tend to q and pc , respectively, and $pc > q$.

Give target A weight

$$a_A s^{w_A - cb_A}, \quad (3)$$

where $a_A > 0$ is a fixed coefficient chosen in advance. A white edge multiplies the weight by s , and a black edge by s^{-c} . Call the sum of all target weights the **potential**, and define the danger of a free edge by

$$D(e) = (1 - s^{-c}) \sum_{A \ni e} a_A s^{w_A - cb_A}.$$

Again Chingiz successively takes p edges of maximum current danger. One black edge decreases the potential by exactly its current danger and can only decrease the dangers of other edges. If the maximum remaining danger after Chingiz's move is D , each of his p chosen edges had danger at least D when chosen, so the potential loss is at least pD . During Bella's q white edges, dangers can grow successively by factors no greater than $1, s, \dots, s^{q-1}$. Hence the increase is at most

$$\frac{s-1}{1-s^{-c}} D(1+s+\dots+s^{q-1}) = \frac{s^q-1}{1-s^{-c}} D \leq pD.$$

Thus the sum of weights (3) does not increase. The incomplete-last-round argument is identical to the one in Lemma I.

Crucially, c may be chosen arbitrarily close to q/p from above. Therefore $1/(c+1)$ may be made arbitrarily close to $p/(p+q)$ from below.

21. The Black Degree of Every Vertex

The targets are the vertex stars: the target for v consists of the $n - 1$ edges incident with v . Take all coefficients $a_v = 1$. The initial potential equals n , so at the end, for every vertex,

$$s^{d_W(v) - cd_B(v)} \leq n.$$

Since $d_W(v) + d_B(v) = n - 1$,

$$d_B(v) \geq \frac{n - 1 - \log_s n}{c + 1}. \quad (4)$$

Dividing (4) by $n - 1$ gives

$$\frac{d_B(v)}{n - 1} \geq \frac{1}{c + 1} - \frac{\log_s n}{(c + 1)(n - 1)}.$$

First choose c close enough to q/p that

$$\frac{1}{c + 1} > \frac{p}{p + q} - \frac{\varepsilon}{2},$$

and then take n large enough that the second term is less than $\varepsilon/2$. This proves part (b). Part (a) follows from the short 1 : 1 case preceding the general lemma.

22. Density Inside Large Sets

Choose c, s as in Lemma II, with

$$\frac{1}{c + 1} > \frac{p}{p + q} - \frac{\varepsilon}{2}.$$

Choose a constant $C = C(p, q, \varepsilon)$ so large that

$$\frac{8}{(c + 1)C \log s} < \frac{\varepsilon}{2},$$

and put

$$m = \lceil C \log n \rceil.$$

For each vertex set U of size $u \geq m$, take all edges inside U as a target and assign coefficient

$$a_U = n^{-2u}.$$

The initial total weight is less than one:

$$\sum_{u=m}^n \binom{n}{u} n^{-2u} \leq \sum_{u=m}^{\infty} n^{-u} < 1.$$

Thus at the end every individual target has weight less than one. With $M = \binom{u}{2}$,

$$n^{-2u} s^{M - (c+1)e_B(U)} < 1,$$

and hence

$$\frac{e_B(U)}{M} > \frac{1}{c + 1} - \frac{2u \log_s n}{(c + 1)M}. \quad (5)$$

The second term is $4 \log_s n / ((c + 1)(u - 1))$. For sufficiently large n , the condition $u \geq m$ gives $u - 1 \geq (C/2) \log n$, so this term is at most

$$\frac{8}{(c + 1)C \log s} < \frac{\varepsilon}{2}.$$

Formula (5) now gives the required density in part (b). For part (a), use the same calculation with the short 1 : 1 lemma.

23. Density Between Large Sets

Choose c, s as in Lemma II so that

$$\frac{1}{c+1} > \frac{p}{p+q} - \frac{\varepsilon}{2}.$$

Then choose $C = C(p, q, \varepsilon)$ so large that

$$\frac{4}{(c+1)C \log s} < \frac{\varepsilon}{2},$$

and put $m = \lceil C \log n \rceil$.

For every ordered pair of disjoint sets (X, Y) , with $|X| = x$, $|Y| = y$, and $x, y \geq m$, take all xy edges between them as a target with coefficient

$$a_{X,Y} = n^{-2(x+y)}.$$

There are at most n^{x+y} pairs of these sizes, so the initial potential is less than

$$\sum_{x,y \geq m} n^{-(x+y)} < 1.$$

As above,

$$n^{-2(x+y)} s^{xy - (c+1)e_B(X,Y)} < 1,$$

and therefore

$$\frac{e_B(X,Y)}{xy} > \frac{1}{c+1} - \frac{2 \log_s n}{c+1} \left(\frac{1}{x} + \frac{1}{y} \right). \quad (6)$$

For $x, y \geq m$, the final term is at most

$$\frac{4}{(c+1)C \log s} < \frac{\varepsilon}{2}.$$

Formula (6) proves part (b). Part (a) is the 1 : 1 specialization stated before the general lemma.

F. Two-objective weight strategies

Weight Lemma III: several target types and q white edges

First, the $1 : 1$ game. If a black edge multiplies the weight of a type- i target by α_i , and a white edge multiplies it by β_i , it is enough that

$$\beta_i - 1 \leq 1 - \alpha_i.$$

Indeed, a black move of maximum danger decreases the potential by at least D , while the following white move increases it by at most D .

Now the general case. In each round Chingiz colors p edges black and Bella then colors q edges white. Every target has a type i . A black target edge multiplies its weight by $\alpha_i \in [0, 1)$, and a white target edge multiplies it by $\beta_i > 1$. Suppose for every type that

$$\beta_i^q - 1 \leq p(1 - \alpha_i). \quad (7)$$

The following table summarizes the balance.

target edge becomes black	target edge becomes white	balance condition
$\times \alpha_i$	$\times \beta_i$	$\beta_i^q - 1 \leq p(1 - \alpha_i)$

Call the sum of all target weights the potential, and define the danger of a free edge by

$$D(e) = \sum_{i, A \ni e} (1 - \alpha_i) w_A.$$

Chingiz's strategy. He successively chooses p free edges of maximum current danger. A black move decreases the potential by exactly the danger of its edge and can only decrease the dangers of other edges. Let the maximum danger after all p choices be D . Each chosen edge had danger at least D when selected, so the potential fell by at least pD .

Estimate Bella's whole next move at once. Suppose a type- i target A contains t of her chosen edges, where $0 \leq t \leq q$, and has weight w_A before her move. Its contribution to the increase is $w_A(\beta_i^t - 1)$. The function $x \mapsto \beta_i^x - 1$ is convex and is zero at $x = 0$, so

$$\beta_i^t - 1 \leq \frac{t}{q}(\beta_i^q - 1) \leq \frac{pt}{q}(1 - \alpha_i).$$

Sum this over all targets and then over the white edges they contain. The danger of each such edge, computed before Bella's move, is at most D . Hence the total increase in potential is at most

$$\frac{p}{q} \sum_{e \text{ chosen white}} D(e) \leq pD.$$

Thus a full round does not increase the potential. An incomplete final Bella move only strengthens the estimate; if Chingiz's last move is incomplete, he takes every remaining edge and Bella does not move again. This proves the lemma. For $q = 1$, it becomes the earlier rule $\beta_i - 1 \leq p(1 - \alpha_i)$.

24. A White Clique and Black Degrees

We give one calculation for the $p : 1$ game. When $p = 1$, it uses the short $1 : 1$ case of Lemma III; for general p , it uses the full lemma.

Put $a = p + 1$ and

$$t = \lceil 2 \log_a n \rceil, \quad k = t + 2.$$

The first target type consists of all k -vertex sets. An alive target with r free edges has weight a^{-r} . Here $\alpha = 0$, $\beta = a$, and (7) is an equality for $q = 1$.

The second target type consists of the vertex stars, with weights

$$n^{-2} s^{d_W(v) - c d_B(v)},$$

where $c > 1/p$ is arbitrarily close to $1/p$, and $s > 1$ is close enough to one that

$$s - 1 \leq p(1 - s^{-c}).$$

Condition (7) also holds for these targets.

The initial weight of the clique targets is at most

$$n^k a^{-k(k-1)/2} \leq a^{-k/2} \leq \frac{1}{an}.$$

Here we used $k-1 \geq 2 \log_a n + 1$, which gives $n \leq a^{(k-2)/2}$.

The initial star weight is $1/n$. Thus the total initial potential is at most $(1+1/a)/n < 1$, and by Lemma III it does not increase. A completely white k -clique would have weight 1, which is impossible. Therefore

$$\omega(W) \leq k-1 = \lceil 2 \log_{p+1} n \rceil + 1.$$

At the end, for every vertex,

$$n^{-2} s^{n-1-(c+1)d_B(v)} \leq \frac{1+1/a}{n},$$

whence

$$d_B(v) \geq \frac{n-1-\log_s((1+1/a)n)}{c+1}.$$

Choose c close enough to $1/p$ from above that

$$\frac{1}{c+1} > \frac{p}{p+1} - \frac{\varepsilon}{2},$$

and then choose n so large that

$$\frac{\log_s((1+1/a)n)}{(c+1)(n-1)} < \frac{\varepsilon}{2}.$$

Then every vertex satisfies

$$d_B(v) \geq \left(\frac{p}{p+1} - \varepsilon \right) (n-1).$$

Part (a) is the case $p = 1$.

25. Two Bounds in the Erdős Game

a)

Choose

$$\frac{1}{2} < c < \frac{1}{\beta} - 1.$$

This is possible precisely because $\beta < 2/3$. Then choose $s > 1$ sufficiently close to one that

$$s-1 \leq 2(1-s^{-c}). \tag{8}$$

Put

$$\Delta = \frac{1}{c+1} - \beta > 0,$$

and choose $C = C(\beta)$ so large that

$$\frac{8}{(c+1)C \log s} < \Delta.$$

Set

$$k = \lceil 2 \log_3 n \rceil + 2, \quad m = \lceil C \log n \rceil.$$

Add two potentials:

1. every alive k -clique has weight 3^{-r} , where r is its number of free edges;
2. every U with $|U| \geq m$ has weight $n^{-2|U|} s^{e_W(U)-ce_B(U)}$.

Bella makes the first move before the ordinary Chingiz-first rounds begin. This move multiplies the first part by at most 3, and the second by at most s . Before that move, the first part is at most

$$\binom{n}{k} 3^{-\binom{k}{2}} \leq n^k 3^{-k(k-1)/2} \leq \frac{1}{3n}.$$

The last inequality is verified exactly as in Problem 24. Before Bella's first move, the second part is at most

$$\sum_{u=m}^n \binom{n}{u} n^{-2u} \leq \sum_{u=m}^{\infty} n^{-u} = \frac{n^{1-m}}{n-1} \leq \frac{1}{n-1}.$$

Thus immediately after Bella's first move the total potential is less than

$$\frac{1}{n} + \frac{s}{n-1} < 1$$

for all sufficiently large n .

Chingiz now moves. For clique targets, $(\alpha, \beta_i) = (0, 3)$, while for targets U , $(\alpha, \beta_i) = (s^{-c}, s)$. The conditions of Lemma III for $p = 2$, $q = 1$ follow from $3 - 1 = 2(1 - 0)$ and (8). Hence Chingiz keeps the total potential below one.

There is no white k -clique, so

$$\omega(W) \leq \lceil 2 \log_3 n \rceil + 1.$$

Moreover, for every U of size $u \geq m$,

$$n^{-2u} s^{\binom{u}{2} - (c+1)e_B(U)} < 1.$$

Therefore

$$\frac{e_B(U)}{\binom{u}{2}} > \frac{1}{c+1} - \frac{4 \log_s n}{(c+1)(u-1)}.$$

By definition of Δ , the first term is $\beta + \Delta$. For large n , $u \geq m$ implies $u-1 \geq (C/2) \log n$, so the second term is at most

$$\frac{8}{(c+1)C \log s} < \Delta.$$

Consequently,

$$e_B(U) > \beta \binom{u}{2},$$

and in particular $e_B(U) \geq \beta \binom{u}{2}$.

b)

Choose a new number

$$\frac{1}{\sqrt{3}} < \beta_0 < \frac{2}{3}.$$

This is legitimate because the conclusion of part (b) does not depend on the number β fixed before part (a). Apply part (a) with β_0 , and let $C = C(\beta_0)$ be the corresponding constant.

Put $m = \lceil C \log n \rceil$. By part (a), every set of at least m vertices has black density at least β_0 . Start with $U_0 = V$. While $|U_i| \geq m$, the average degree in $B[U_i]$ is at least

$$\frac{2e_B(U_i)}{|U_i|} \geq \beta_0(|U_i| - 1).$$

Thus $B[U_i]$ contains a vertex v_i of degree at least $\beta_0(|U_i| - 1)$. Let U_{i+1} be its neighbors within U_i . The chosen vertices are pairwise adjacent.

If $r_i = |U_i|$ and $a = \beta_0/(1 - \beta_0)$, then

$$r_{i+1} \geq \beta_0(r_i - 1), \quad r_{i+1} + a \geq \beta_0(r_i + a).$$

Inductively,

$$r_i + a \geq \beta_0^i(n + a).$$

Put

$$t = \left\lfloor \frac{\log((n+a)/(m+a))}{\log(1/\beta_0)} \right\rfloor.$$

By the definition of t , $\beta_0^t(n+a) \geq m+a$. Also, for $i \leq t$, $\beta_0^i \geq \beta_0^t$. Hence $r_i \geq m$ throughout the first t selections. The process really does build a clique of at least t vertices, and

$$\omega(B) \geq t \geq \frac{\log((n+a)/(m+a))}{\log(1/\beta_0)} - 1 = \frac{\log n - \log \log n - O(1)}{\log(1/\beta_0)}. \quad (9)$$

The condition $\beta_0 > 1/\sqrt{3}$ says

$$\log(1/\beta_0) < \frac{1}{2} \log 3.$$

Thus the coefficient of $\log n$ in (9) is strictly greater than $2/\log 3$, whereas

$$\omega(W) \leq \frac{2 \log n}{\log 3} + O(1).$$

The strict gap between the leading $\log n$ terms dominates the $O(\log \log n)$ loss. Therefore $\omega(B) > \omega(W)$ for every sufficiently large n .

26. The $p : q$ Game and the Golden Ratio

Put $A = p + 1$. The second hypothesis of the problem gives

$$\frac{1}{\log(1 + q/p)} > \frac{2q}{\log A}. \quad (10)$$

By continuity, choose ρ slightly smaller than $p/(p + q)$ so that still

$$\frac{1}{\log(1/\rho)} > \frac{2q}{\log A}. \quad (11)$$

This ρ will be the guaranteed black density in every large set.

Since $\rho < p/(p + q)$, we have $q/p < 1/\rho - 1$. Choose

$$\frac{q}{p} < c < \frac{1}{\rho} - 1,$$

and then choose $s > 1$ close enough to one that

$$s^q - 1 \leq p(1 - s^{-c}). \quad (12)$$

Such an s exists because, after division by $s - 1$, the two sides tend to q and $pc > q$ as $s \downarrow 1$.

Put

$$\Delta = \frac{1}{c + 1} - \rho > 0,$$

and choose C so large that

$$\frac{8}{(c + 1)C \log s} < \Delta.$$

Next set

$$k = \lceil 2q \log_A n \rceil + 2, \quad m = \lceil C \log n \rceil.$$

Track two target types simultaneously.

1. For each k -vertex set, take all its internal edges. While the target is alive, meaning it has no black edge, give it weight $A^{-r/q}$, where r is its number of free edges. A black edge multiplies the weight by 0, and a white edge by $A^{1/q}$.
2. For each U with $|U| \geq m$, use weight

$$n^{-2|U|} s^{e_W(U) - ce_B(U)}.$$

A black edge inside U multiplies this weight by s^{-c} , and a white edge by s .

For the first target type, (7) is an equality:

$$(A^{1/q})^q - 1 = A - 1 = p;$$

for the second type, it is exactly (12). Thus Weight Lemma III applies after each Chingiz move of p edges and the following Bella move of q edges.

We must still account for Bella moving first in the problem statement. Before her first move, the total weight of the first type is at most

$$\binom{n}{k} A^{-\binom{k}{2}/q} \leq n^k A^{-k(k-1)/(2q)} \leq A^{-k/(2q)} \leq \frac{A^{-1/q}}{n}.$$

Bella's first q white edges increase this sum by a factor of at most A . The initial weight of the second type is at most

$$\sum_{u=m}^n \binom{n}{u} n^{-2u} \leq \sum_{u=m}^{\infty} n^{-u} \leq \frac{1}{n-1},$$

and Bella's first move increases it by a factor of at most s^q . Therefore, immediately after her first q edges, the total potential is at most

$$\frac{A^{1-1/q}}{n} + \frac{s^q}{n-1} < 1$$

for sufficiently large n . Chingiz now moves, and Lemma III keeps the total potential below one until the end.

A completely white k -clique would have weight 1, so

$$\omega(W) \leq k-1 = \lceil 2q \log_{p+1} n \rceil + 1. \quad (13)$$

On the other hand, for every U of size $u \geq m$, its type-2 weight is less than one. Hence

$$n^{-2u} s^{\binom{u}{2} - (c+1)e_B(U)} < 1,$$

which gives

$$\frac{e_B(U)}{\binom{u}{2}} > \frac{1}{c+1} - \frac{4 \log_s n}{(c+1)(u-1)}.$$

For sufficiently large n , $u \geq m$ implies $u-1 \geq (C/2) \log n$. By the choice of C , the final term is less than Δ , and so

$$e_B(U) > \rho \binom{u}{2}. \quad (14)$$

Repeat the greedy construction from Problem 25(b). Start with $U_0 = V$. While $|U_i| \geq m$, choose in $B[U_i]$ a vertex of degree at least $\rho(|U_i| - 1)$, and replace U_i by its neighborhood. The chosen vertices form a black clique. If $r_i = |U_i|$ and $a_0 = \rho/(1-\rho)$, then

$$r_{i+1} + a_0 \geq \rho(r_i + a_0).$$

Consequently the process makes at least

$$\left\lfloor \frac{\log((n+a_0)/(m+a_0))}{\log(1/\rho)} \right\rfloor$$

selections, and

$$\omega(B) \geq \frac{\log n - \log \log n - O(1)}{\log(1/\rho)}. \quad (15)$$

By (11), the coefficient of $\log n$ in (15) is strictly greater than $2q/\log(p+1)$, while (13) gives

$$\omega(W) \leq \frac{2q}{\log(p+1)} \log n + O(1).$$

The strict gap in the leading coefficients dominates the $O(\log \log n)$ loss. Hence for all sufficiently large n ,

$$\omega(B) > \omega(W).$$

Remark on the golden ratio. The first hypothesis of the problem formally follows from the second. Indeed,

$$(1 + q/p)^{2q} \geq (1 + 1/p)^2,$$

so $p+1 > (1 + 1/p)^2$. After multiplication by p^2 , this is equivalent to

$$(p+1)(p^2 - p - 1) > 0,$$

that is, $p > \varphi$. The golden ratio is the exact boundary when $q = 1$; the proof itself needs only the second, stronger hypothesis.

27*. The exact Erdős conjecture

Statement. Bella and Chingiz play on the edges of K_n . Bella moves first and colors one uncolored edge white on every move; after each of her moves, Chingiz colors two uncolored edges black. When all edges have been colored, Chingiz wins if $\omega(B) > \omega(W)$, and Bella wins otherwise.

Prove or disprove that Chingiz has a winning strategy for every $n \geq 4$.

Open problem. In accordance with the project statement, no solution is given here.

Addendum 2: dynamics and tournaments

28. Rock, Paper, Scissors on a Graph

Extended problem by Mark Pimenov, 2026. The original tree part is Problem 11.5 for Grade 11 from the 2024 final round of the Higher School of Economics “Vysshaya Proba” Olympiad.

Number the objects modulo 3 so that A_{i+1} defeats A_i .

a) A tree

Idea. On a tree, the colors can be lifted uniquely from residues modulo 3 to integer heights such that adjacent heights differ by at most 1. In one minute every vertex then simply takes the greatest height in its closed neighborhood.

Choose a root. Give its color any integer representative. Moving away from the root, assign to every vertex v an integer $h(v)$, congruent modulo 3 to the number of its object, so that for every edge uv

$$h(v) - h(u) \in \{-1, 0, 1\}.$$

Once the height at the root has been chosen, this can be done uniquely: a tree has no cycle that could impose contradictory requirements.

If a neighbor has height $h(v) + 1$, it carries precisely the object that defeats the object at v , so the height of v increases by 1. If no such neighbor exists, its height does not change. Thus after one minute

$$h_{t+1}(v) = \max(\{h_t(v)\} \cup \{h_t(u) : u \sim v\}). \quad (1)$$

The new heights of adjacent vertices still differ by at most 1, so they really are a lift of the new coloring.

Iterating (1) gives

$$h_t(v) = \max\{h_0(u) : d(u, v) \leq t\}. \quad (2)$$

Let $M = \max h_0$. After finitely many minutes, the ball on the right-hand side of (2) contains a vertex of height M , for every v . Hence all heights eventually become M . From that moment on all objects are equal and the process stops changing.

b) The exact time

Answer. The largest possible number of minutes is

$$\boxed{n - 1}.$$

Formula (2) also gives the exact answer for every initial position. If

$$Z = \{u : h_0(u) = M\},$$

then stabilization takes

$$T = \max_v d(v, Z)$$

minutes. On a tree with n vertices, $T \leq n - 1$.

Equality is attained on the path $v_1 v_2 \dots v_n$. Put

$$h_0(v_1) = \dots = h_0(v_{n-1}) = 0, \quad h_0(v_n) = 1.$$

In other words, v_n carries A_1 , while every other vertex carries A_0 . The boundary between the two objects advances by one edge per minute, and v_1 changes only in minute $n - 1$.

c) A graph containing a cycle

Idea. A coloring of a cycle has an integer winding number, which is invariant under the process. Its value is zero in any stationary coloring.

Choose a simple cycle

$$C = v_0 v_1 \dots v_{\ell-1} v_0.$$

For the directed edge $v_i v_{i+1}$, let $d_i(t) \in \{-1, 0, 1\}$ be the unique integer congruent modulo 3 to the difference between the object numbers at v_{i+1} and v_i . Let $\varepsilon_i(t)$ be 1 if the object at v_i changes in that minute, and 0 otherwise.

Then

$$d_i(t+1) = d_i(t) + \varepsilon_{i+1}(t) - \varepsilon_i(t). \quad (3)$$

The right-hand side does not need to be reduced modulo 3. For example, if $d_i = 1$, then v_i sees the object that defeats its own and must change, so the right-hand side is 0 or 1. The case $d_i = -1$ is symmetric. Summing (3) around the cycle shows that

$$D(C) = \sum_{i=0}^{\ell-1} d_i(t)$$

is invariant.

If $\ell = 3$, place A_0, A_1, A_2 around the cycle. If $\ell \geq 4$, put

$$A_0, A_1, A_2, A_0$$

at v_0, v_1, v_2, v_3 , respectively, and put A_0 at every other vertex of the cycle. Objects outside the cycle may be chosen arbitrarily. In both cases three successive differences equal 1 and all other differences equal 0, so $D(C) = 3$.

If all changes had stopped, the endpoints of every edge would carry the same object: for two distinct objects, one defeats the other, and the loser must change. Thus a stationary coloring of a connected graph is monochromatic and has $D(C) = 0$. This contradicts invariance. Therefore changes continue forever.

d) An odd number of objects

Let $m = 2r + 1$. On a tree, again lift each object number to an integer height, but now require on every edge that

$$h(v) - h(u) \in \{-r, -r+1, \dots, r\}.$$

This representative of the difference modulo m is unique. A vertex changes exactly when it has a neighbor of greater height; when it changes, its height increases by 1.

After the simultaneous update, all edge differences remain in the same interval. Indeed, if one endpoint of an edge is higher, the lower endpoint must increase, so the difference cannot cross r or $-r$. The global maximum M never changes, while all heights are nondecreasing and at most M . Consequently, the integer potential

$$\Phi = \sum_v (M - h(v))$$

strictly decreases in every minute in which some change occurs. Hence on a tree the process stops after finitely many minutes.

For a graph containing a cycle, define

$$d_i(t) \in \{-r, -r+1, \dots, r\}$$

on its cycle edges exactly as above. Equation (3) remains valid as an equality of integers, so $D(C) = \sum d_i$ is invariant. Prescribe the differences on three successive cycle edges to be

$$r, \quad r, \quad 1,$$

and all remaining differences to be 0. Their sum is $2r+1 = m$, so they define a consistent assignment of objects around the cycle. It has $D(C) = m \neq 0$, whereas every stationary assignment has $D(C) = 0$. Thus under the generalized rule as well, every graph containing a cycle admits an infinite process.

29. Two Leagues

Problem by Mark Pimenov, 2026.

For a team v , define its final score balance in the whole tournament by

$$s(v) = \#\{\text{wins by } v\} - \#\{\text{losses by } v\}.$$

The sum of all balances is zero: every match contributes +1 to the winner and -1 to the loser.

a) At least one half

After the teams have been assigned, let A and B be the sets of teams in the First and Second Leagues. In $\sum_{v \in A} s(v)$, every match between two First-League teams cancels internally: one end-point contributes $+1$, the other -1 . Only matches between the leagues remain. Therefore

$$\begin{aligned} \sum_{v \in A} s(v) &= \#\{\text{First-League wins over the Second League}\} \\ &\quad - \#\{\text{Second-League wins over the First League}\}. \end{aligned} \tag{4}$$

Strategy. On every turn, the First League chooses a remaining team with the largest value of $s(v)$. Pair each First-League turn with the following Second-League turn. If the two teams chosen in such a pair are a_i and b_i , then

$$s(a_i) \geq s(b_i),$$

because b_i was still available when a_i was chosen. Summing over all pairs gives

$$\sum_{a \in A} s(a) \geq \sum_{b \in B} s(b).$$

The total sum is zero, so the left-hand side is nonnegative. By (4), the First League wins at least half of all matches between the leagues.

b) When a strict majority can be guaranteed

Answer. Exactly when

$$n \equiv 2 \pmod{4}.$$

Write $n = 2m$. There are exactly m^2 matches between the two leagues. If m is odd, this number is odd, so an exact half is impossible. The strategy from part (a) guarantees at least one half and therefore guarantees strictly more than one half. This is precisely the case $n \equiv 2 \pmod{4}$.

It remains to construct an obstruction when $n \equiv 0 \pmod{4}$, that is, when m is even. Start with a regular tournament on $2m + 1$ teams: arrange them cyclically and let team i defeat the next m teams. Every team has exactly m wins. Delete one team. Of the remaining $2m$ teams, exactly m have balance $+1$, and the other m have balance -1 .

Because m is even, pair the $+1$ teams in advance, and separately pair the -1 teams. After every First-League choice, the Second League chooses the partner of that team. Each pair is split evenly between the two leagues, so the two leagues have the same total balance, namely zero. By (4), their numbers of wins against one another are equal. Hence the First League cannot guarantee a strict majority.