Contents

Chapter 1

Grids and tables

In Olympiad combinatorics, especially in olympiads like IZhO, All-Russian, and Saint-Petersburg, very often encounters problems with grids. In this chapter, I have tried to write every single idea or method that I have encountered during my preparation for olympiad mathematics. Let us start with the simplest one:

1.1 Invariants

Invariants are the most popular and the most efficient method to crack grid problems, but invariants might be very tricky and extremely non-obvious. This thought will become clearer through numerous examples in this chapter. As usual, we will start with the simplest one.

1.1.1 Invariants as a method of solution

The most popular type of problem involving the grids and tables are formulated in the following way:

> You have some board (not necessary rectangular) and you need to divide/tile it into some pieces (not necessary same type or rectangular by shape). Can you do it?

or like this

> You have some board (again, not necessary rectangular) and you need to place/move figure(s) on the board, so that some criteria are met.

In this types of problems it is useful to color the board in such a way that some properties are constant – they are *invariant*.

One of such problems is the following one

Example 1.1.1
Prove that a 10×10 *board cannot be cut along the grid lines into* 1×4 *rectangles.*

In this problem the thing we want to preserve – is the content of each 1×4 rectangle.

Solution 1:

Firstly, to not overcomplicate things, let's try to use as few color as possible – 2. With such a limitation in colors natural and obvious thing to attach to is the following *invariant*:

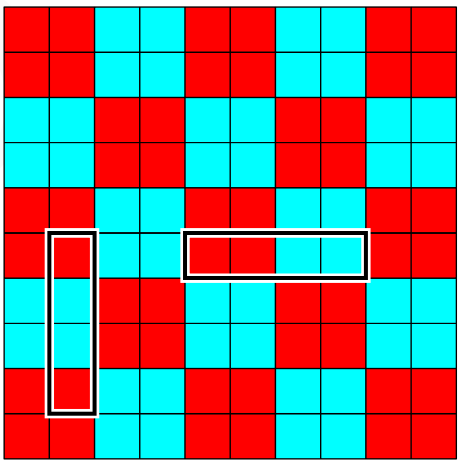
$\rightarrow$ *Every* 1×4 *rectangle contains 2 of red and cyan cells.*

With this idea in the mind we need to create the coloring for which this is true. We will construct it dynamically (pretend that we are artist coloring the canvas as we go), start by doing the most general thing you can:

Place the 1×4 in the corner of the table and paint 2 cell red and other 2 of cyan. Now shift your 1×4 rectangle by 1, new rectangle already has 2 cyan cells and 1 red cell, so the remaining cell must be red.

Continuing this logic we will get to one line of the table colored. But that is only one dimension, we need 2.

Fortunately, simply by rotating obtained coloring and sometimes inverting the colors we get to the following:



Now the board has 52 red cells and 48 cyan ones, i.e. not an equal number.

This means that it will not be possible to cut the 10×10 board into 1×4 tetrominoes.

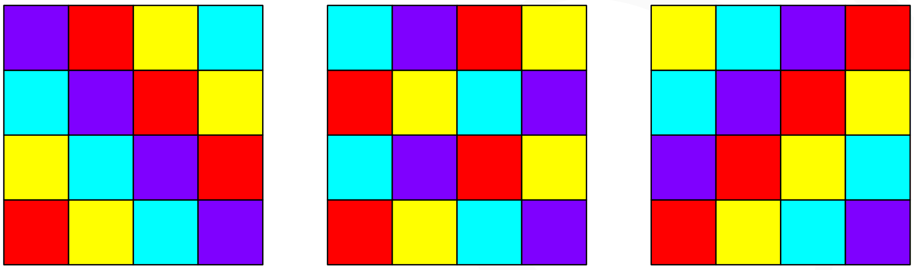
Solution 2:

As you may have noticed, after first 4 cells (in any direction) pattern repeats itself, so we can just focus on the one 4×4 board.

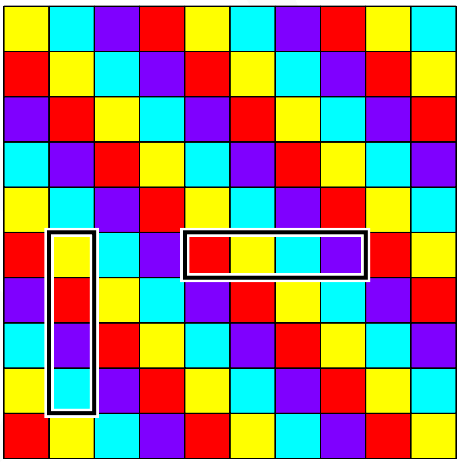
As you have the 4-celled figure you also may be tempted to use 4 colors, so if we change our invariant to the following:

→ *Every* 1×4 *rectangle contains one cell of each of the 4 colors.*

You can easily get to one of multiple working examples.



Now using the same logic as in the previous example, we can use one of such 4×4 tables as a blueprint and spam it to get this:



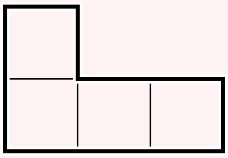
With this coloring, the board has 25 cells of the 1st and 3rd colors, 26 cells of the 2nd, and 24 cells of the 4th, i.e., not same number.

This means that it will not be possible to cut the 10×10 board into 1×4 tetrominoes.

This idea of coloring as we go and slightly changing the shape of our focus is the main idea for constructing such colorings.

In the previous example we encountered with the simplest colorings and invariants. The next example with the similar taste

Example 1.1.2
Is it possible to cover a 10×10 *board with the following pieces without them overlapping?*
Note: The pieces ***can*** *be flipped and/or turned.*

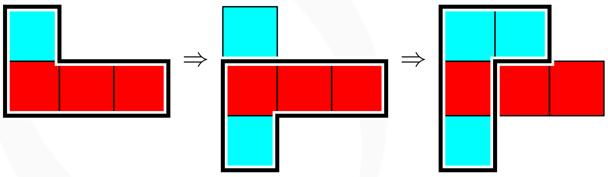


This problem is a little tricky one, to understand why, try using previous method and obtain coloring such that every figure contains exactly 2 red and 2 cyan cells (or the same but with 4 colors).

There is an issue with this approach – only possible coloring is checkerboard pattern, which is simply useless in this problem.

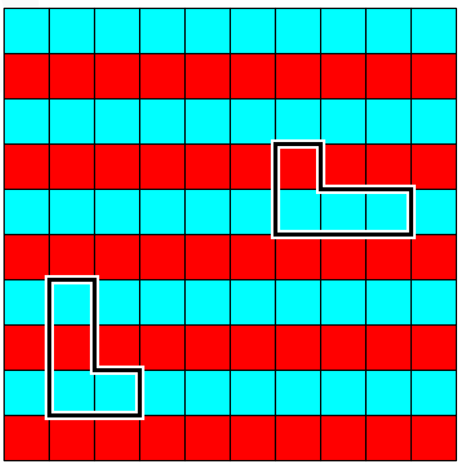
So we need to find a little more complex coloring:

Try coloring such that every figure has 3 cyan and 1 red cell:



You fail on the third figure, so try to color the board in such a way that every figure has either 1 or 3 red cells.

Continuing the logic you get to the following coloring.



There are 50 cyan squares and 50 red squares. We call the figure cyan if it covers 3 cyan squares, and red if it covers three red squares.

The number of cyan figure is equal to the number of red figure. This tells us that the total number of figure must be even. This would mean that the number of squares should be divisible by 8. Since there are 100 squares, there is no possible cover.

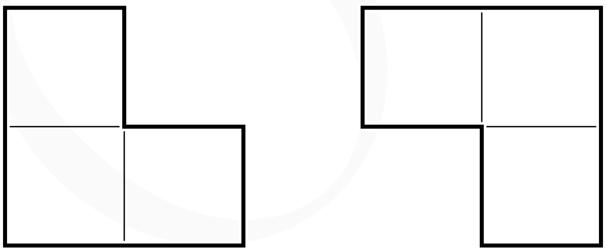
Example 1.1.3 (ISL 2023 C1)
Let m and n be positive integers greater than 1. In each unit square of an $m \times n$ grid lies a coin with its tail side up. A move consists of the following steps.

1. *select a 2×2 square in the grid;*
2. *flip the coins in the top-left and bottom-right unit squares;*
3. *flip the coin in either the top-right or bottom-left unit square.*

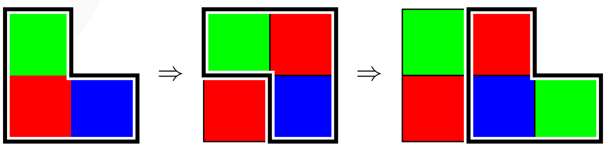
Determine all pairs (m, n) for which it is possible that every coin shows head-side up after a finite number of moves.

Answer: $\boxed{1}$.

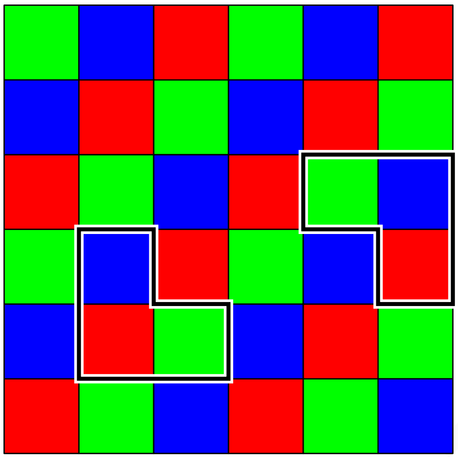
If we do changes within the moves simultaneously, then there are only 2 types of moves:



And, as in the first example, we want to make a coloring such that every move is *invariant.* So we color every cell of figure in unique color:

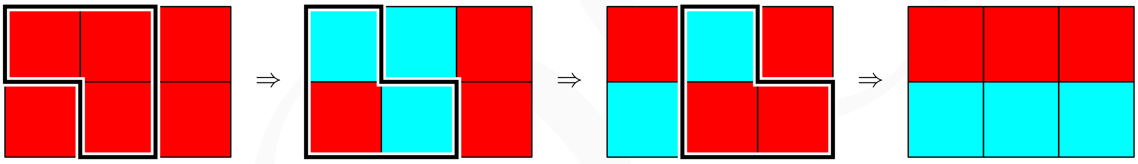


Continuing this we get to the following coloring:



Note, that in every move one of each type of coin is flipped. Therefore, the amount of every color initially should have been the same. This is only achieved when $3 \mid mn$, so the bound is done.

For construction, it is easy to see we can always change the state of 2×3 rectangle. Now, the only thing we need to do is learn how to flip all the coins in a 1×3 array. This however can be done in the following way:

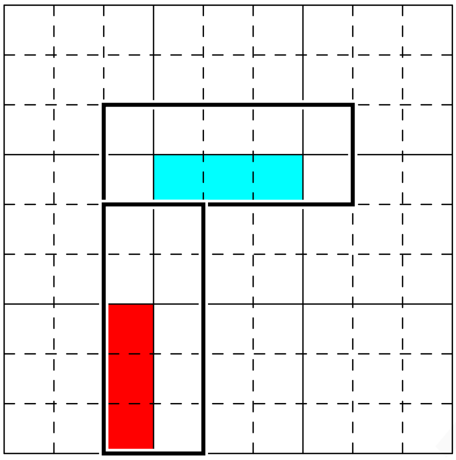


However, invariants are not only about colorings; rather, invariants about cuts and divisions. Next problem illustrates that idea more clearly.

Example 1.1.4
A 60×60 checkered square is divided into 2×5 tiles. Prove that it is possible to partition the square into 1×3 rectangles such that each 2×5 tile contains at least one rectangle entirely.

Problems wants us to cut everything into 1×3 rectangles, one of the ways to do so is cut everything into 3×3 squares and divide each of them into 1×3 rectangles after.

Draw a marking that divides the original 60×60 square into 3×3 squares.



Note that each 2×5 tile has a side of length 5 divided by this marking into either two parts (their lengths are 2 and 3) or three parts (with lengths 1, 3, 1).

Therefore, on each tile there is a 1×3 rectangle that is not divided into parts by the marking lines (the figure shows examples of such rectangles), we will paint one such rectangle at each point. Obviously, the painted rectangles do not intersect, since they lie in different tiles.

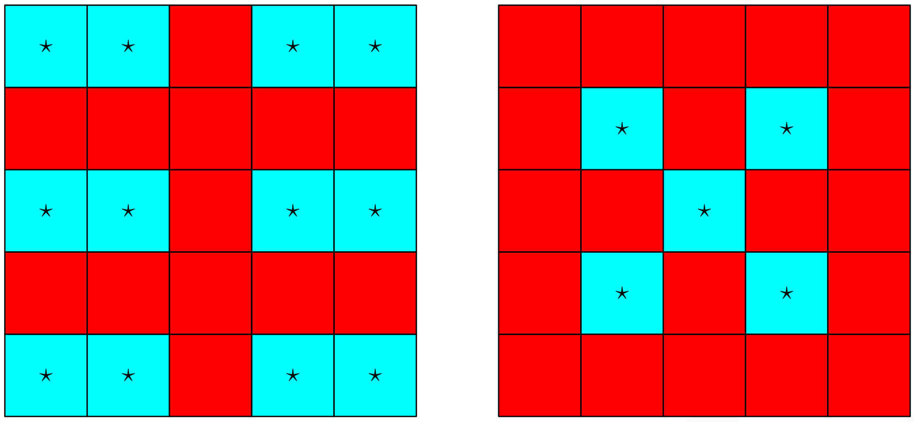
If in some 3×3 square more than one 1×3 rectangle is painted, then they lie in the same direction (all horizontally or all vertically), so they do not intersect.

Therefore, each square of the 3×3 marking can be divided into 1×3 rectangles in such a way that all the painted rectangles will belong to this division.

It might seem that invariant problems are kinda easy: just spam one of the basic colorings or cuts. However, invariant problems frequently can be extremely non-obvious and tricky, with mind-blowing colorings or cuts. In next 2 examples that idea will be clearer.

Example 1.1.5 (APMO 2007)
A regular 5×5 array of lights is defective, so that toggling the switch for one light causes each adjacent light in the same row and in the same column as well as the light itself to change state, from on to off, or from off to on. Initially all the lights are switched off.
After a certain number of toggles, exactly one light is switched on. Find all the possible positions of this light.

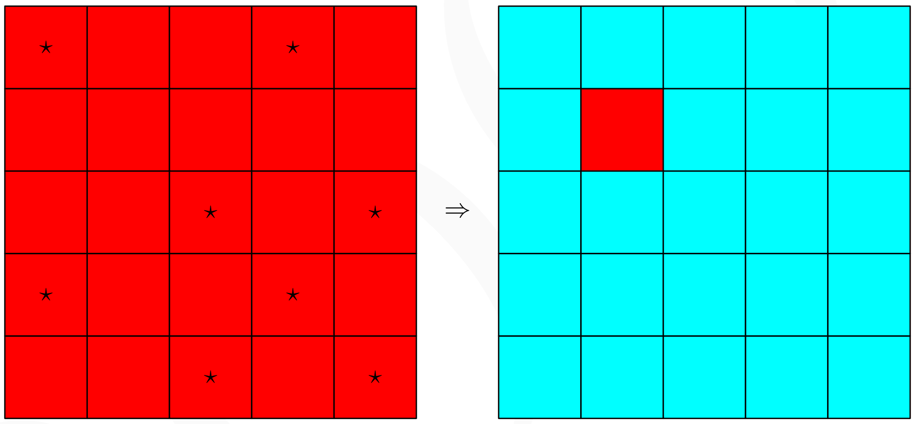
Consider the following coloring:



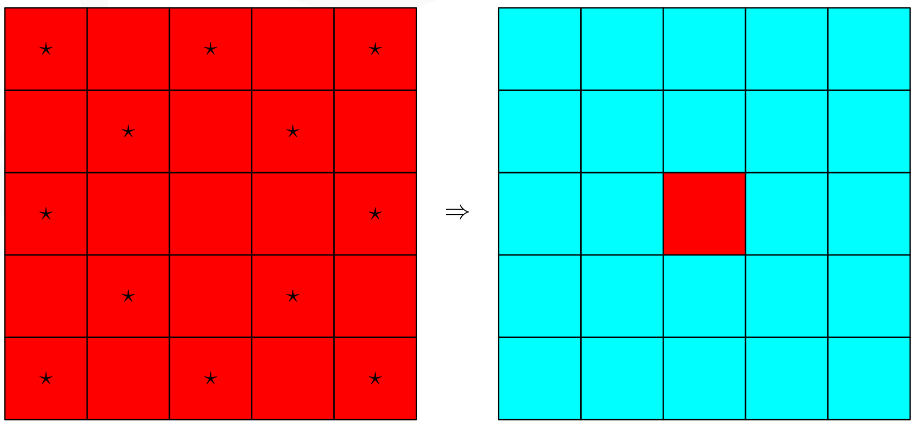
There are 12 cyan cells, and it is easily verifiable that any move on the board will affect 0 or 2 of these cells. Thus none of these can remain alone.

Rotating this coloring by 90 degrees eliminates all except for the 5 points.

Example for $(2, 2)$ cell, star is the cells that we switch:



Example for the middle cell:

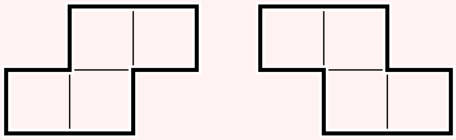


Example 1.1.6 (ISL 2014 C4)

Construct a tetromino by attaching two 2×1 dominoes along their longer sides such that the midpoint of the longer side of one domino is a corner of the other domino. This construction yields two kinds of tetrominoes with opposite orientations. Let us call them $S-$ and $Z-$tetrominoes, respectively.

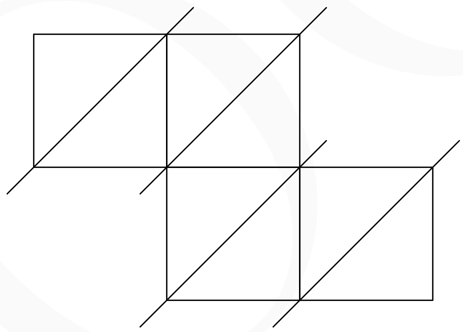
Assume that a lattice polygon P can be tiled with $S-$ tetrominoes. Prove that no matter how we tile P using only $S-$ and $Z-$tetrominoes, we always use an even number of $Z-$tetrominoes.

Note: tetrominoes cannot be rotated.

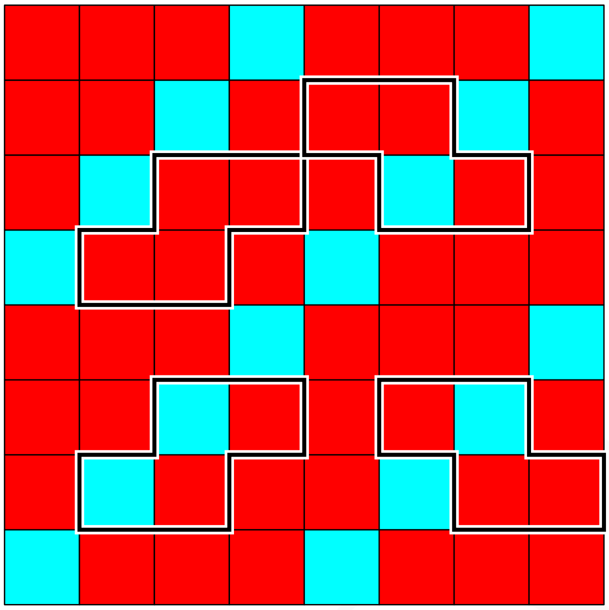


$S-$tetromino $Z-$tetromino

One of the main differences between $S-$ and $Z-$tetrominoes is the fact that every cell in the $Z-$tetromino lies on unique (/) diagonal. We will abuse this by making a diagonal coloring.



Color some cells of grid cyan and red, as a repeating of coloring below.



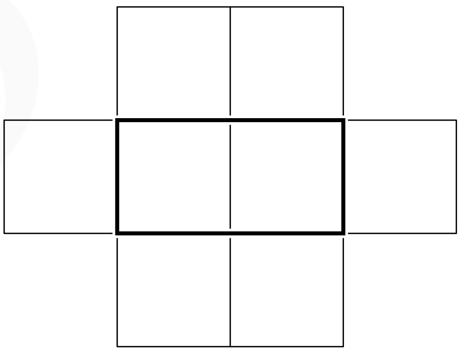
Observe that any $S-$tetromino covers even number of cyan cells, so P also cover even number of cyan. But any $Z-$tetromino covers an odd number of cyan cells, so the conclusion follows.

1.1.2 Invariants as a method of counting or bounding

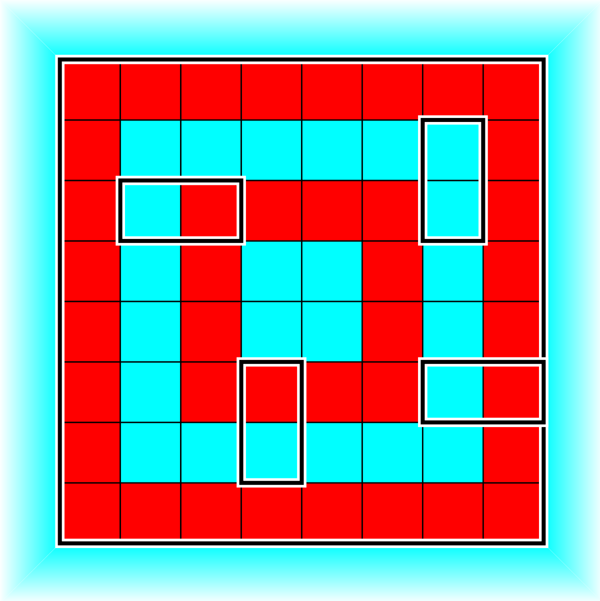
Example 1.1.7 (EGMO 2019)
Given a natural number n, find the largest possible number of dominoes that can be placed on a $2n \times 2n$ checkered board so that any cell is adjacent to or belongs to exactly one domino.

Bound:

Every domino actually represents the following region:



But the outer squares can be cut off if the domino lies on the outer region of the board. This motivates us to imaginary "color" extra space around board in one color, and continuing this logic color the table in layers as follows:



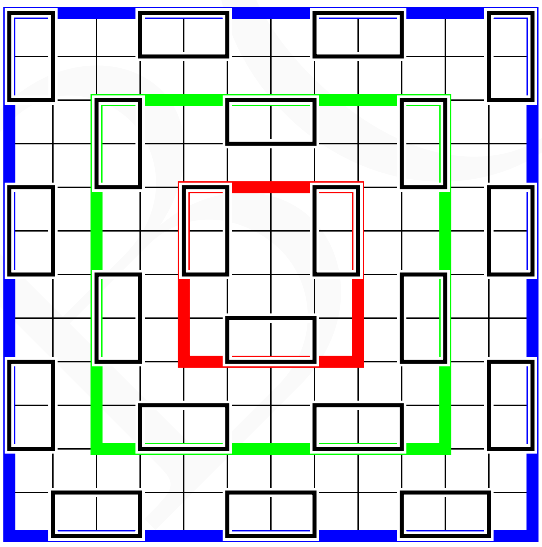
Note that every domino covers or is adjacent to exactly 4 red cells. So the maximum number of dominoes $= \frac{\text{number of red cells}}{4}$.

If $2 \mid n \implies$ number of red cells $= 3+7+\cdots+(2n-1) = \frac{(2n-1)+3}{2} \cdot \frac{n}{2} = \frac{n(n+1)}{2}$.

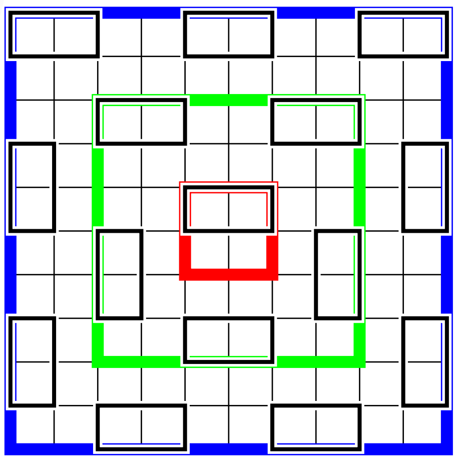
If $2 \nmid n \implies$ number of red cells $= 1+5+\cdots+(2n-1) = \frac{(2n-1)+1}{2} \cdot \frac{n+1}{2} = \frac{n(n+1)}{2}$.

So the number of dominoes $= \frac{n(n+1)}{8}$

Example:



for even n



for odd n

Example 1.1.8 (All-Russian 2023)
Let n be an odd integer. In a $2n \times 2n$ board, we color $2(n-1)^2$ cells. What is the largest number of three-square corners that can surely be cut out of the uncolored figure?

Answer: $\boxed{1}$.

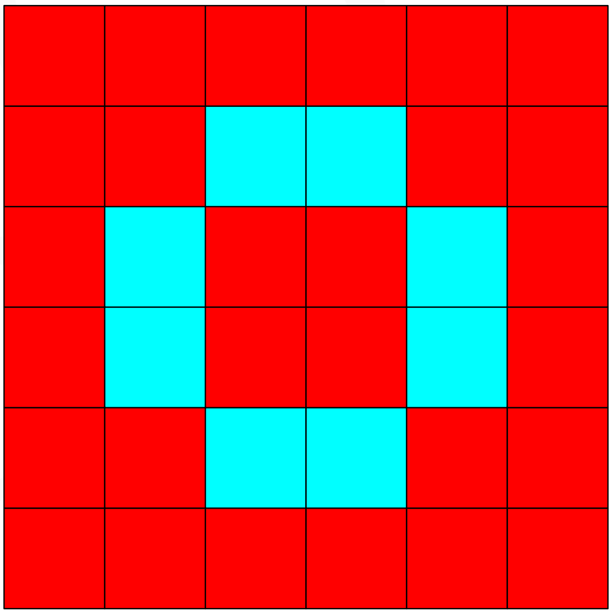
Bound:

Divide the grid into 2×2 squares. In order to stop a three corner from being cut out from one of these squares, there must be at least two squares shaded. Notice, there can only by $(n-1)^2$ such squares with at least two squares shaded. Thus, we can cut out three corners form at least $n^2 - (n-1)^2 = 2n - 1$ of them.

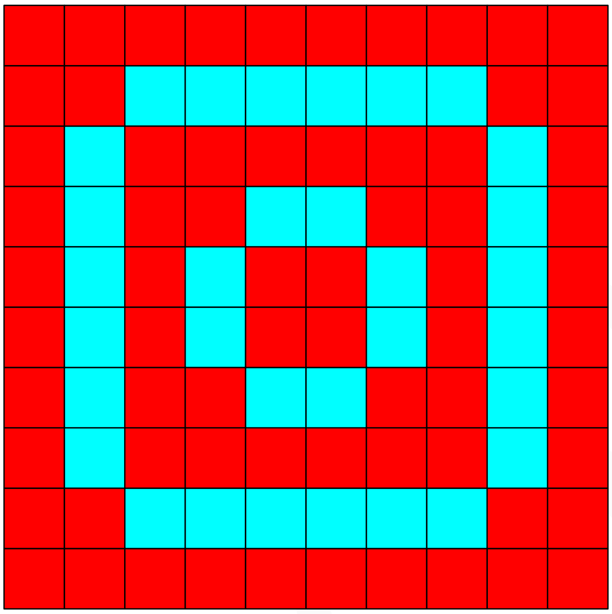
Construction:

We will proceed by induction. For $n = 3$, by bound we *should* be able to place only 5 three-square corners. so we need to distribute 5 empty 2×2 squares in such a way, that we cannot place any additional three-square corners.

Note that if we have 2 of empty 2×2 squares next to each other, it is very hard to avoid extra three-square corners from appearing, so trying to avoid it we get to the following construction:



When going from $n \to n+2$ since we are using induction it makes sense to use previous construction to our advantage, so placing a copy for the construction for $2n \times 2n$ in the center of the $2n+2 \times 2n+2$ grid feel natural. Then with the additional $2(n+1)^2 - 2(n-1)^2 = 4n$ squares surround the central $2n \times 2n$ construction. Then you are able to cut out 4 additional three corner pieces from the corners in addition to the $2n-1$ three corners from the center.



Example 1.1.9 (239 Olympiad 2022)
A chip is placed in the lower left-corner cell of the 15×15 *board. We can move to the cells that are adjacent to the sides or the corners of its current cell. We must also alternate between horizontal and diagonal moves the first move must be diagonal What is the maximum number of moves it can make without stepping on the same cell twice*

Answer: $\boxed{1}$.

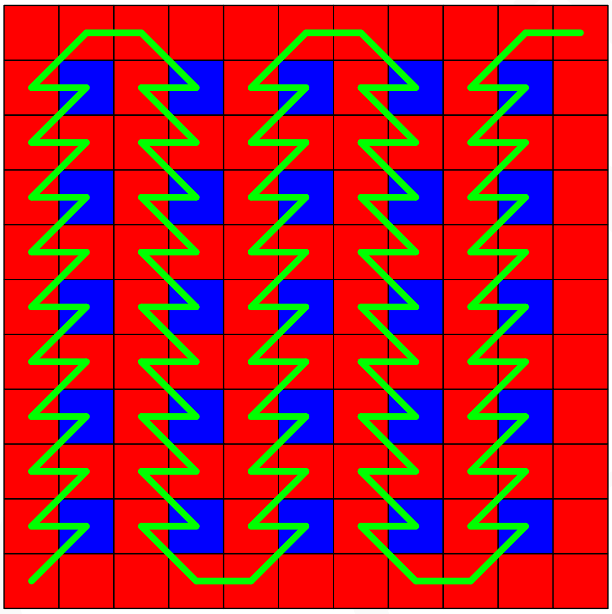
Bound:

For the motivation, notice that we can change vertical position only by moving diagonally, so after 2 moves you change your horizontal position by 0 or 2, and your vertical position by 1. So to get to your initial position you must spend al least 4 moves. From this point you should consider all positions that you can get to after even number of moves. And from that point notice pattern in every 4th move.

Let's paint all 49 intersections of rows and columns with even numbers cyan. After the first move, the chip will end up in a cyan cell. After any 4 moves in a row from a cyan cell, the chip will end up on a cyan cell again. Thus, having made $1 + 49 \cdot 4 = 197$ moves, the chip will have to visit $1 + 49 = 50$ cyan cells. Such a number of cyan cells does not exist, so it will not be possible to make more than 196 moves.

Example:

First, the chip (moving up) goes around the two left columns without two cells, then the next two columns without two cells (this time moving down), then the next two columns, etc. At the end of the path, the chip will visit one cell from the last column. In total, it will visit 14 cells in each column except the last one, and one more cell in the last column, a total of 197 cells, i.e., it will make 196 moves.



1.2 Graphs

Example 1.2.1
In the 100×100 *grid* k *cells are colored in black color. If at any moment 3 of the 4 cells that lie as the corners of rectangle with sides parallel to sides of the initial square are black, after a minute the fourth cell will also become black. What is the minimum* k*, such that after some time the whole square will turn black?*

Answer: $\boxed{1}$.

First idea in graphs is

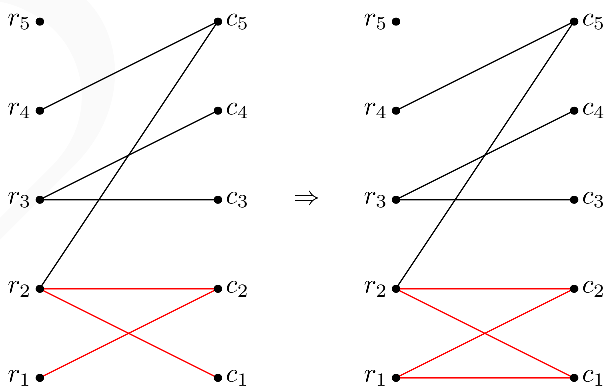
Turning table into bipartite graph by assigning a vertex to each row and column and drawing an edge between them if corresponding intersection of row and column is colored.

Because of the way we set up rows and columns this idea works well if we are working with something related to horizontal or vertical movements. And this problem is perfect for demonstrating it.

So, formally:

Let us construct a bipartite graph with vertices $r_1, \ldots, r_{100}$ corresponding to the rows of the board, and vertices $c_1, \ldots, c_{100}$ corresponding to its columns. We connect vertices r_i and c_j by an edge if the cell at the intersection of the corresponding row and column is black.

So the move now consists of completing the cycle of length 4.



Notice that connectivity remains constant. That is, if any two vertices were connected before, they remain connected. And if not, then after the move they still remain unconnected.

As the final graph is connected, the initial should also be. So the total number of edges must be no less than 199 (cuz the tree).

Example 1.2.2 (All-Russian 2013)
Petya put several chips on a 50×50 *board, no more than one in each cell. Prove that Vasya has a way to put no more than 99 new chips (possibly none) on the free squares of this same board so that there is still no more than one chip in each cell, and each row and each column of this board has an even number of chips.*

Construct the same bipartite graph as in previous example, but we connect vertices r_i and c_j by an edge if the cell is empty. Then Vasya's goal is reformulated as follows: it is required to mark no more than 99 edges so that an even number of unmarked edges emerge from each vertex. Indeed, if Vasya places chips in the cells corresponding to the marked edges, then an even number of free cells will remain in each row and each column.

We will prove a more general fact: in any graph on $n \geq 2$ vertices, it is possible to mark no more than $n-1$ edges so that an even number of unmarked edges emerge from each vertex.

Induction on n. The base ($n = 2$) is obvious.

The induction step. Let $n > 2$. If the graph contains a vertex of degree 0, then it suffices to throw it away and apply the induction hypothesis.

If there is a vertex of degree 1, then we can mark the only edge emanating from it, throw it away together with this edge, and apply the induction hypothesis to the remaining graph.

Now let the degree of each vertex be at least 2. Let us leave an arbitrary vertex along an edge, leave the vertex we arrived at along another edge, and so on; this process can be continued until we return to a vertex we have already visited. Thus, a cycle has been found in the graph. Throwing its edges out of the graph does not change the parities of the vertex degrees; therefore, we can just erase those edges. Applying the same process to it, sooner or later we will obtain a graph in which the degree of some vertex does not exceed 1; and for such graphs the assertion has already been proved.

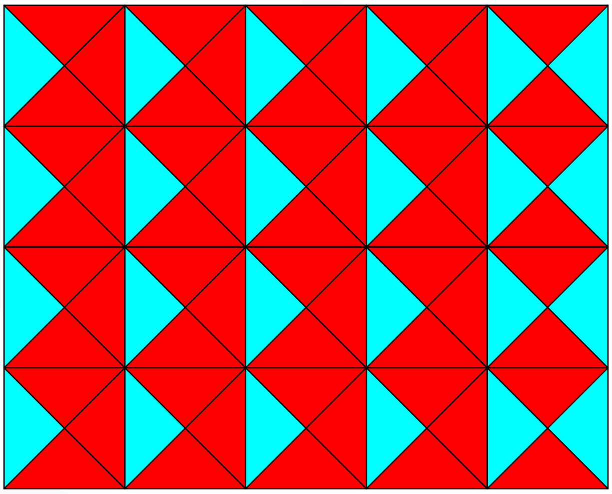
1.3 Double counting

Example 1.3.1 (Saint Petersburg 2017)
In each cell of an $m \times n$ rectangle, two diagonals were drawn, resulting in the rectangle being divided into $4mn$ triangles. All triangles were painted black or white so that each white triangle has a common side with at least one black triangle. What is the smallest number of black triangles that could be in such a coloring?

Answer: $\boxed{1}$.

Example:

WLOG assume that table has m rows and n columns, where $m < n$. In each cell we paint the left triangle black, and in each cell of the right column we paint another right triangle. In total we get $mn + m$ black triangles, and it is easy to see that this example satisfies the condition of the problem.



Solution:

Bound:

Consider any row of a rectangle. It contains $2n$ "vertical" triangles (triangles with a vertical side). If such a triangle is painted white, draw an arrow leading from it to the adjacent black triangle (it lies in the same row). If the vertical triangle is black, draw an arrow from it to itself.

We drew $2n$ arrows in total. Note that no more than two arrows lead to each triangle in this row: arrows can lead to a horizontal triangle only from two vertical triangles in the same cell, and to a vertical triangle only from itself and the vertical triangle adjacent to it. Therefore, there are at least n black triangles in this row. Similarly, there are at least m black triangles in each column.

Let's go further. Note that if there are exactly n black triangles in some row, then exactly two arrows enter each of them. In particular, the leftmost vertical triangle is not painted black (otherwise the arrow would lead to it only from itself), so it is white. This means that the arrow from it leads to one of the neighboring horizontal triangles, i.e. this horizontal triangle is black. But then another arrow must lead to it, so the right vertical triangle in this cell is also white. Continuing in the same way, we find that in each cell of this row there is exactly one black triangle, and it is horizontal. Similarly, if in some column there are exactly m black triangles, then all of them are vertical, and there is one in each cell.

However, a row with exactly n black triangles and a column with exactly m black triangles cannot exist simultaneously: the cell at their intersection would contain exactly one black triangle, which must be both horizontal and vertical. In other words, either in each row there are at least $n+1$ black triangles, or in each column there are at least $m+1$ black triangles. In one case, there are no less than $m(n+1) = mn+m$ black triangles in total, in the other, no less than $n(m+1) = mn+n$, in any case, no less than $mn + \min(m, n)$.

Example 1.3.2 (All-Russian 2017)
Each cell of a 100×100 board is painted either black or white, and all cells adjacent to the board's border are black. It turns out that there is no single-color 2×2 checkered square anywhere on the board. Prove that there is a 2×2 checkered square on the board whose cells are painted in a checkerboard pattern.

Solution:

FTSoC assume that there are no single-colored or checkerboard-colored 2×2 squares on the board.

Consider all the grid segments separating two cells of different colors (let's call them separators); let their number be N.

In any 2×2 square, there is either exactly one cell of one color and three cells of the other, or two adjacent white cells and two adjacent black cells. In both cases, there are exactly two separators inside the square.

There are 99^2 2×2 squares in total, and each separator lies inside exactly two of them (since the separators do not adjoin the border).

Therefore, $N = \frac{2(99^2)}{2} = 99^2$. In each row and each column, the first and last cell are black; therefore, there must be an even number of color changes, so the total number of separators must be even. Contradiction.

Example 1.3.3 (BMO SL 2019)
Determine the largest natural number N having the following property: every 5×5 array consisting of pairwise distinct natural numbers from 1 to 25 contains a 2×2 sub-array of numbers whose sum is, at least, N.

Answer: $\boxed{1}$.

First, enumerate the columns and rows. Then, we will select all possible $3^2 = 9$ choices for the odd column with an odd row.

By collecting all such pairs of an odd column with an odd row, we double count some squares. Indeed, we took some 3^2 squares 5 times, some 12 squares 3 times, and there are some 4 squares (namely all the intersections of an even column with an even row) that we did not take in such pairs.

It follows that the maximum total sum over all 32 choices of an odd column with an odd row is $5 \times (17 + 18 + \cdots + 25) + 3 \times (5 + 6 + \cdots + 16) = 1323$.

So, by averaging argument, there exists a pair of an odd column with an odd row $\frac{1323}{9}$ with sum at most $= 147$.

Then all the other squares of the array will have sum at least $(1 + 2 + \cdots + 25) - 147 = 178$. However, for these squares there is a tiling with $2 \times 2 arrays$, which are 4 in total. So there is an 2 x 2 array, whose numbers have a sum at least $\frac{178}{4} > 44$. So, there is a 2×2 array whose numbers have a sum at least 45. This argument gives that $k_{max} \geq 45$.

Here, we have a bound that $k_{\max} \geq 45$, but now we should prove that $k = 45$ is optimal. Fortunately, this part looks less impressive than the previous part: we can do this by providing an example:

23	13	19	12	24
7	2	9	4	5
18	16	17	15	21
8	1	10	3	6
22	14	20	11	25

1.4 Induction

Example 1.4.1

On an $n \times n$ chart where $n \geq 4$, stand n "+" signs in cells of one diagonal and a "–" sign in all the other cells. In a move, one can change all the signs in one row or in one column, (– changes to + and + changes to –). Prove that it is impossible to reach a stage where there are fewer than n pluses on the board.

Solution:

Note that operating twice on a row is equivalent to not operating on it at all. So we can assume that each row and column has been operated upon 0 or 1 times. Now we use induction on n.

The base case $n = 4$ is not entirely trivial, but is left to the reader in keeping with my general habit of dismissing base cases.

Now passing to the induction step, given an $n \times n$ board there are at least $(n-1)$ pluses in the bottom right $(n-1) \times (n-1)$ square by the induction hypothesis. If we have a plus in the first row or column we are done. Suppose there is no plus in the first column or row. Then either the first row or the first column (but not both) has been operated upon (otherwise the top left square would have a plus).

WLOG the first row has been operated upon. Then columns $2, 3, \ldots, n$ have all been operated upon (otherwise row 1 would have a plus). Also no other row has been operated upon (otherwise the first column would have a plus). But in this case, the lower right $(n-1) \times (n-1)$ square has had all its columns and none of its rows operated upon, and hence each column has $(n-2)$ pluses. In total it has $(n-2)(n-1) > n$ pluses, so in this case as well we are done.

1.5 Complex numbers

Example 1.5.1 (IMO 2016)
Find all integers n for which each cell of $n \times n$ table can be filled with one of the letters I, M and O in such a way that:

- *in each row and each column, one third of the entries are I, one third are M and one third are O*
- *on any diagonal, if the number of entries on the diagonal is a multiple of three, then one third of the entries are I, one third are M and one third are O.*

We claim that $9 \mid n$ is the only answer.

For $n = 9$, consider the following table

I	I	I	M	M	M	O	O	O
M	M	M	O	O	O	I	I	I
O	O	O	I	I	I	M	M	M
I	I	I	M	M	M	O	O	O
M	M	M	O	O	O	I	I	I
O	O	O	I	I	I	M	M	M
I	I	I	M	M	M	O	O	O
M	M	M	O	O	O	I	I	I
O	O	O	I	I	I	M	M	M

Bound:

Since each of I, M, O appears the same number of times, $3 \mid n^2$ and so $3 \mid n \implies n = 3k$.

Tile the board with 3×3 sub-grids. For each 3×3 grid call the central square its representative cell. Assign the numbers $1, \omega$ and ω^2 to I, M and O, respectively.

Add the numbers assigned to all rows and columns indexed with numbers $\equiv 2 \pmod 3$ and all numbers assigned to the diagonals with number of entries a multiple of 3. This total sum must be zero since each of the terms is zero for a row and a column and any diagonal having number of elements a multiple of 3.

However, the total sum is the sum of all entries of the $n \times n$ board and the sum of all numbers assigned to the representative cells. As the sum of all entries is zero, it follows that the sum of all numbers assigned to the representative cells is zero. Let x of them be labeled I, y of them be labeled M, z of them be labelled O. We have $x + y + z = k^2$ and $x + y\omega + z\omega^2 = 0$.

There are two ways to conclude that $x = y = z$ from here.

Notice that multiplying the last equation by ω and ω^2 gives $z+x\omega+y\omega^2 = 0$ and $y + z\omega + x\omega^2 = 0$. Considering $1, \omega, \omega^2$ as constants and x, y, z as parameters, we see that this system of equations has a solution if and only if

$$\begin{vmatrix} x & y & z \\ z & x & y \\ y & z & x \end{vmatrix} = \frac{1}{2}(x + y + z)\left((x - y)^2 + (y - z)^2 + (z - x)^2\right) = 0.$$

As $x + y + z = k^2 > 0$, we have $x = y = z$.

Alternatively, note that the polynomial $1 + x + x^2$ is irreducible and has no double roots. The polynomial $x + yx + zx^2$ has a common root with $1 + x + x^2$ so if $1+x+x^2 \nmid x+yx+zx^2$ then $x-\omega$ is a polynomial with integer coefficients. This is clearly false, so the former must hold and we have $x = y = z$.

Thus, $3 \mid x + y + z = k^2$ and $3 \mid k$. It follows that $9 \mid n$.

1.6 Problems

Problem 1.6.1 (All-Russian 2016) 1950 dominoes were cut out from 100×100 sheet of paper. Prove that it is possible to cut out a four-cell figure from the remaining part.

(If such a figure is already among the remaining parts, it is considered that it was cut out.)

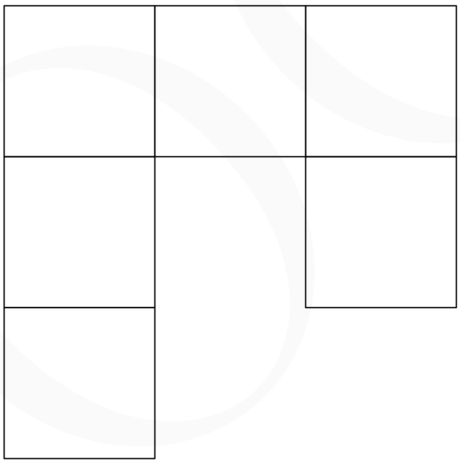
Problem 1.6.2 (EGMO 2016) Let m be a positive integer. Consider a $4m \times 4m$ array of square unit cells. Two different cells are *related* to each other if they are either in the same row or in the same column. No cell is related to itself. Some cells are colored blue, such that every cell is related to at least two blue cells. Determine the minimum number of blue cells.

Problem 1.6.3 On a 11×11 checkered board, 22 squares are marked so that exactly two squares are marked on each vertical and each horizontal. Two arrangements of marked squares are equivalent if, by changing the verticals and horizontals between each other any number of times, we can obtain the other arrangement from one. How many nonequivalent arrangements of marked squares are there?

Problem 1.6.4 (Italian TST 1995) An 8×8 board is tiled with 21 trominoes (3×1 tiles), so that exactly one square is not covered by a tromino. No two trominoes can overlap and no tromino can stick out of the board. Determine all possible positions of the square not covered by a tromino.

Problem 1.6.5 (IMO 1993) On an infinite chessboard, a solitaire game is played as follows: at the start, we have n^2 pieces occupying a square of side n The only allowed move is to jump over an occupied square to an unoccupied one, and the piece which has been jumped over is removed. For which n can the game end with only one piece remaining on the board?

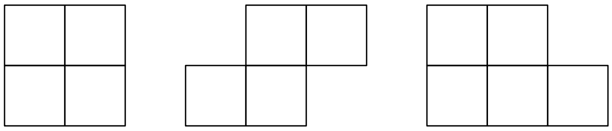
Problem 1.6.6 (IMO 2004) Define a "hook" to be a figure made up of six unit squares as shown below in the picture, or any of the figures obtained by applying rotations and reflections to this figure.



Determine all $m \times n$ rectangles that can be covered without gaps and without overlaps with hooks such that

- the rectangle is covered without gaps and without overlaps
- no part of a hook covers area outside the rectangle.

Problem 1.6.7 (Vietnam 1993) With the following shapes we tile a 1993×2000 board. Let s be the number of shapes used of the first two types. Find the largest possible value of s.



Problem 1.6.8 (Romania 2007) In an $n \times n$ board the squares are painted black or white. Three of the squares in the corners are white and one is black. Show that there is a 2×2 square with an odd number of white unit squares.

Problem 1.6.9 (USAJMO 2023) Consider an n-by-n board of unit squares for some odd positive integer n. We say that a collection C of identical dominoes is a maximal grid-aligned configuration on the board if C consists of $(n^2-1)/2$ dominoes where each domino covers exactly two neighboring squares and the dominoes don't overlap: C then covers all but one square on the board. We are allowed to slide (but not rotate) a domino on the board to cover the uncovered square, resulting in a new maximal grid-aligned configuration with another square uncovered. Let $k(C)$ be the number of distinct maximal grid-aligned configurations obtainable from C by repeatedly sliding dominoes. Find the maximum value of $k(C)$ as a function of n

Problem 1.6.10 (IZhO 2024) We are given $m \times n$ table tiled with 3×1 stripes and we are given that $6 \mid mn$. Prove that there exists a tiling of the table with 2×1 dominoes such that each of these stripes contains one whole domino.

Problem 1.6.11 (ISL 2010) 2500 chess kings have to be placed on a chessboard so that

- no king can capture any other one (i.e. no two kings are placed in two squares sharing a common vertex);
- each row and each column contains exactly 25 kings.

Find the number of such arrangements. (Two arrangements differing by rotation or symmetry are supposed to be different.)

Problem 1.6.12 (ISL 2023) Let $n \geqslant 2$ be a positive integer. Paul has a $1 \times n^2$ rectangular strip consisting of n^2 unit squares, where the i^{th} square is labelled with i for all $1 \leqslant i \leqslant n^2$. He wishes to cut the strip into several pieces, where each piece consists of a number of consecutive unit squares, and then translate (without rotating or flipping) the pieces to obtain an $n \times n$ square satisfying the following property: if the unit square in the i^{th} row and j^{th} column is labelled with a_{ij}, then $a_{ij} - (i + j - 1)$ is divisible by n.

Determine the smallest number of pieces Paul needs to make in order to accomplish this.

Problem 1.6.13 (ISL 2022) Let n be a positive integer. A Nordic square is an $n \times n$ board containing all the integers from 1 to n^2 so that each cell contains exactly one number. Two different cells are considered adjacent if they share a common side. Every cell that is adjacent only to cells containing larger numbers is called a valley. An uphill path is a sequence of one or more cells such that:

- the first cell in the sequence is a valley,
- each subsequent cell in the sequence is adjacent to the previous cell,
- the numbers written in the cells in the sequence are in increasing order.

Find, as a function of n, the smallest possible total number of uphill paths in a Nordic square.

Problem 1.6.14 (All-Russian 2023) A 100×100 square is divided into 2×2 squares. It is then broken into dominoes. What is the smallest number of dominoes that could be inside the squares of the partition?

Problem 1.6.15 (All-Russian 2016) A 100×100 grid is given, the cells of which are painted black and white. In all columns there are equal numbers of black cells, while in all rows there are different numbers of black cells. What is the maximum possible number of pairs of adjacent multi-colored cells?

Problem 1.6.16 (Tuymaada 2021) In a $n \times n$ table ($n > 1$) k unit squares are marked. One wants to rearrange rows and columns so that all the marked unit squares are above the main diagonal or on it. For what maximum is it always possible?

Problem 1.6.17 (All-Russian 2013) 400 three-cell corners (rotated as desired) and another 500 cells were cut out of 55×55 checkered square along the borders of the cells.

Prove that some two cut out figures have common boundary segments.

Problem 1.6.18 (IZhO 2020) Some squares of a $n \times n$ table ($n > 2$) are black, the rest are white. In every white square we write the number of all the black squares having at least one common vertex with it. Find the maximum possible sum of all these numbers.

Problem 1.6.19 (USAJMO 2025) Let m and n be positive integers, and let

$\mathcal{R}$ be a $2m \times 2n$ grid of unit squares.

A domino is a 1×2 or 2×1 rectangle. A subset S of grid squares in $\mathcal{R}$ is domino-tileable if dominoes can be placed to cover every square of S exactly once with no domino extending outside of S. Note: The empty set is domino tileable.

An up-right path is a path from the lower-left corner of $\mathcal{R}$ to the upper-right corner of $\mathcal{R}$ formed by exactly $2m + 2n$ edges of the grid squares.

Determine, with proof, in terms of m and n, the number of up-right paths that divide $\mathcal{R}$ into two domino-tileable subsets.

Chapter 2

Algorithms

Algorithms are vast part of olympiad combinatorics. It might seem that algorithms are not so important, or even algorithms are full intuition, but this is only part of the story. In this chapter, we will present crucial ideas for cracking complicated algorithm problems. However, as usual, we will start with the most intuitive approach to crack algorithms.

2.1 How to construct own algorithms?

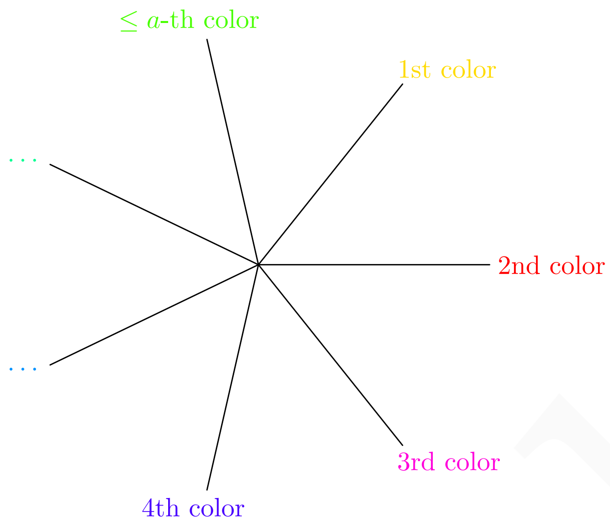
2.1.1 Greedy algorithms

Greedy algorithms are algorithms that make the best possible short – term choices, hence in each step maximizing short – term gain. These words will seem clearer in next chapter:

Example 2.1.1
In a graph G with n vertices, there is no vertex with degree greater than a. Show that one can color the vertices using at most $a+1$ colors, such that no two neighboring vertices of the same color.

Solution:

We will provide clear algorithm to this problem. At first, we will arrange the vertices in an arbitrary order. Let the colors be $1, 2, 3\ldots,$ and color the first vertex with color 1. Here is a greedy part of our algorithm: in each stage, take the next vertex in the order and color it with the smallest color that has not yet been used on any of its neighbors.



Now we will prove that our algorithm indeed works. Clearly this algorithm ensures that two adjacent vertices will not be the same color. It also ensures that at most $a+1$ colors are used: each vertex has at most a neighbors, so when coloring a particular vertex v, at most a colors have been used by its neighbors, so at least one color in the set $\{1, 2, 3, \ldots, a+1\}$ has not been used. The minimum such color will be used for the vertex v. Hence, all vertices are colored using colors in the set $\{1, 2, 3, \ldots, a+1\}$ and the problem is solved.

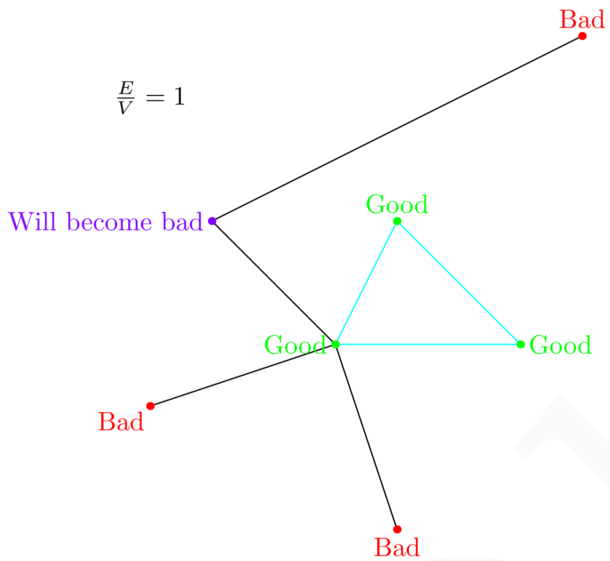
To recall, in this problem we employed the simplest, almost foolish algorithm - yet it proved remarkably effective. We have only one obstacle, and we have tried to overcome only this obstacle in the most effective way, without paying attention to what happens to others. Of course, this approach will not be convenient, if we have several important quantities. However, greedy algorithms might be surprisingly helpful, even in complex problems, like next two examples.

Example 2.1.2

In a graph G with V vertices and E edges, show that there exists an induced subgraph H with each vertex having degree at least $\frac{E}{V}$.
(In other words, a graph with average degree d has an induced subgraph with minimum degree at least $\frac{d}{2}$).

Solution:

Note that the average degree of a vertex is $\frac{2E}{V}$. Intuitively, we should remove the "bad" vertices: vertices that have degree $< \frac{E}{V}$. Thus, a natural algorithm for finding such a subgraph is as follows: start with graph G, and as long as there exists a vertex with degree $< \frac{E}{V}$, delete it.



However, remember that while deleting a vertex we are also deleting the edges incident to it, and in the process vertices that were initially not 'bad' may become bad in the subgraph formed. What if we end up with a graph with all vertices bad? Fortunately, this will not happen: notice that the ratio of $\frac{\text{edges}}{\text{vertices}}$ is strictly increasing (it started at $\frac{E}{V}$ and each time we deleted a vertex, less than $\frac{E}{V}$ edges were deleted by the condition of our algorithm).

Hence, it is impossible to reach a stage when only one vertex is remaining, since in this case the $\frac{\text{edges}}{\text{vertices}}$ ratio is 0. So, at some point, our algorithm must terminate, leaving us with a graph with more than one vertex, all of whose vertices have degree at least $\frac{E}{V}$.

Example 2.1.3 (ISL 2001)
A set of three nonnegative integers x, y, z with $x < y < z$ satisfying $z - y, y - x = a, b$, where a and b are distinct positive integers and $a < b$, is called a historic set. Show that the set of all nonnegative integers can be written as a disjoint union of historic sets.

Solution:

Note that a historic set is in the form $\{x, x + a, x + a + b\}$ (which we will call a small set) or $\{x, x + b, x + a + b\}$ (which we will call a large set).

We can use the following algorithm to achieve this task:

We let x be the smallest number not yet covered. If none of the numbers in the small set have been covered already, then we use a small set, otherwise we use a large set.

Now we prove that this algorithm works. We prove that if the small set does not work, then the large set will always work.

Assume for the sake of contradiction that the large set does not work.

Clearly the x element can not have already been covered.

If the $x+a+b$ element has been covered, then that means that on the previous step, we either used the small set on $x+b$ or the large set on $x+a$. However, this is impossible as $x < x+a < x+b$.

If the $x+b$ has already been covered, then either we used the small set on $x+b-a$, or we used any set on $x-a$. Clearly the first one is impossible as $x+b-a > x$. Now we consider the second case. If we used the small set on $x-a$, then it would imply that we would have already covered x, a contradiction. If we used the large set, then that means that some element in the small set was already covered. If $x+b$ was already covered, then this implies that x was already covered, contradiction. Otherwise, this implies that x was already covered, another contradiction.

Therefore, a situation where both sets do not work, and the algorithm always works

Example 2.1.4 (239 Olympiad 2017)
An invisible tank is on a 100×100 table. A cannon can fire at any 60 cells of the board after that the tank will move to one of the adjacent cells (by side). Then the process is repeated. Can the cannon grantee to shoot the tank?

Answer: $\boxed{1}$.

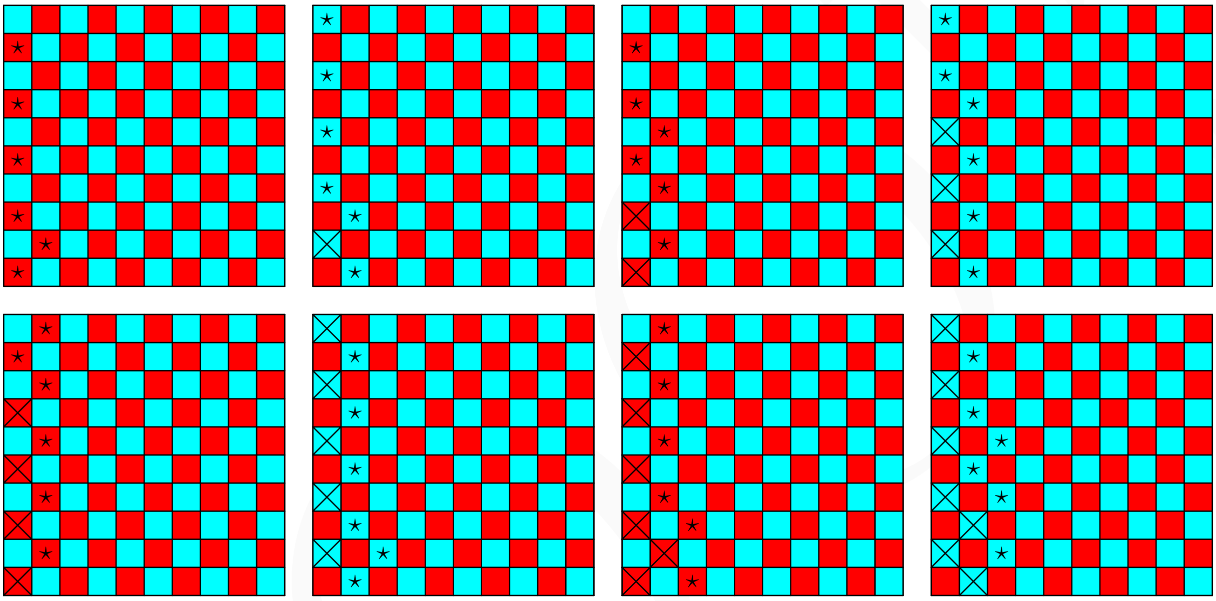
Solution:

First of all, for fresh eye, there is no connection between 100 and 60. Therefore, an idea for generalization is located nearby. Indeed, it is possible to solve this problem for $2n \times 2n$ table and with $n+1$ cells in each shoot. We will present solution for this generalization of problem.

Okay, the tank moves to one of the adjacent cells after each cannon shooting, so there should be something that can regulate and bound the displacement of the tank. For an experienced eye, checkerboard coloring might seem a great option. Indeed, after each canon shooting, tank changes its color, so we can eliminate at least half of the cells in our table. We will discuss more about such strategies in next chapter!

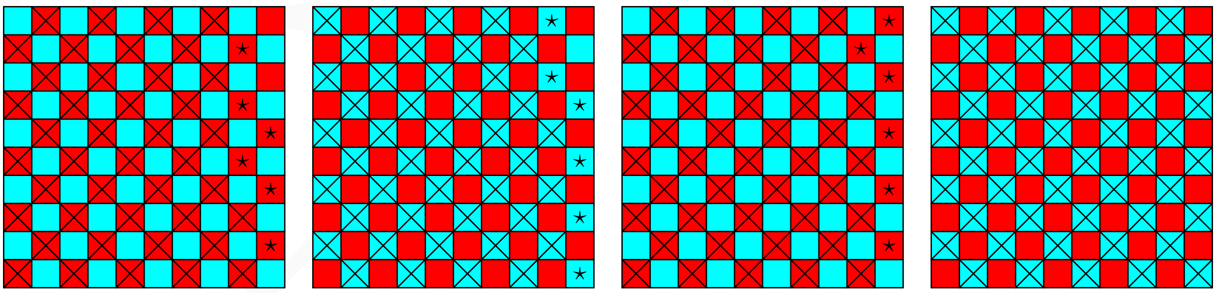
Let's back to our algorithm. Because we should guaranty, we should eliminate every possibility of tank hiding, and this is our greedy approach: we can check other cells, only if we make sure that there is no tank in previous cells.

With this idea in mind and other basic instruments of combinatorics, it is not hard to construct this algorithm that is depicted below. We presented algorithm for $n = 5$, but elaboration of this algorithm for $n = 50$ is obvious.



After each move tank changes the color of the cell it is in, so we change the cells that we shoot.

Here are 4 last moves for $n = 5$:



After some (finite) amount of moves every cell of one color (in our case – cyan) is marked with cross. So if tank was not hit by the cannon, then our original assumption (tank is on the red) was wrong. But we can just repeat the process with the opposite color and now surely hit the tank.

2.1.2 Pairings or divide and conquer

Example 2.1.5

Example 2.1.6 (Russia 2002)
There is one red cell and $k(k > 1)$ blue cells, as well as a deck of $2n$ cards numbered from 1 to $2n$. Initially, the entire deck lies in the red cell in an arbitrary order. From any cell, you may take the top card and move it either to an empty cell or onto a card whose number is exactly one greater. What is the largest n for which it is possible, using such operations, to move the entire deck into one of the blue cells?

Solution:

Let us construct an example showing that if $n > k$, it is impossible.

Suppose the cards (from top to bottom) are initially arranged so that first come all the odd cards (in arbitrary order), and then all the even cards, with the very top of them being card number $2n$. Then the first k moves are uniquely determined: the odd cards are placed into the free positions. The next move, if $n > k$, is impossible; and if $n = k$, the only possible move is to place card number $2n - 1$ back into the original pile, which is pointless because we return to the previous position. Therefore, in this arrangement, the pile cannot be transferred.

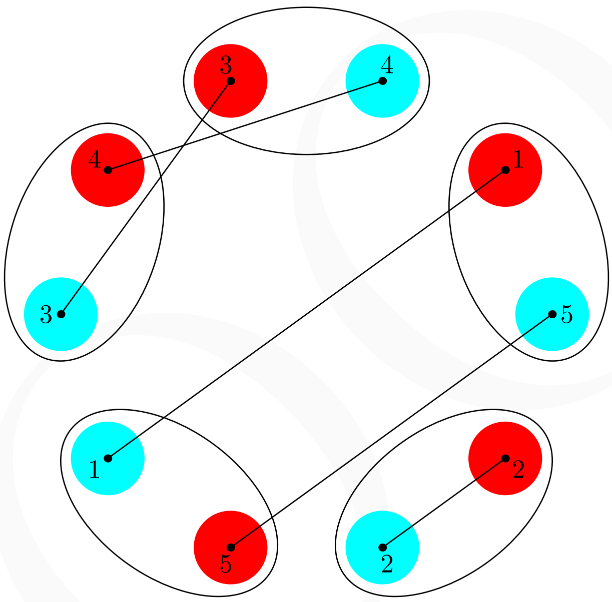
Now suppose $n < k$. We show how the transfer can be organized.

Divide all the cards into pairs $(1, 2), (3, 4), \ldots, (2n - 1, 2n)$ and assign to each pair a separate empty cell (there will be at least one cell left unassigned; we call it the free cell). Now we attempt to place each card taken from the top of the red pile into its assigned cell. This may fail only if the card has number $2i$, while card $2i - 1$ is already in that cell; in that case we can move card $2i$ into the free cell, then place card $2i - 1$ on top of it, assign this cell to the pair, and declare the previously assigned cell to be the new free one. In this way, eventually all the cards are distributed among the cells in pairs. After that, using the free cell, it is easy to collect them into a single deck in the correct order.

Example 2.1.7 (All-Russian 2005)
There are 100 *representatives of* 50 *countries sitting at the rounded table, two from each country. Prove that they can be divided into two groups in such a way that each group will have one representative from each country, and each person was in the same group with no more than one of his neighbors.*

Solution:

At first, we will label 100 people by numbers from 1 to 100 clockwise, and let's pair them in groups of two consecutive guys: $\{1, 2\}, \{3, 4\}, \ldots, \{99, 100\}$.



Now we will present algorithm to solve our problem:

Color 1 black, then color his countryman white, then color 1's countryman's neighbor black, then his countryman white, and so on, until we reach 2, whom we color white. We have colored several of our pairs completely, and we can initiate another process, starting from another pair.

2.2 Induction + Recursion

In the previous chapter, we have considered the simplest algorithms, without any additional ideas. In this chapter, we will look through various examples that will deal with the reference to smaller cases.

Example 2.2.1
Cards numbered 1 to n are arranged at random in a row with $n \geq 5$. In a move, one may choose any block of consecutive cards whose numbers are in ascending or descending order and switch the block around.
For example, if $n = 9$, then (9 1 $\underline{6\ 5\ 3}$ 2 7 4 8) might be changed to (9 1 $\underline{3\ 5\ 6}$ 2 7 4 8).
Prove that in at most $2n - 6$ moves, one can arrange the n cards so that their numbers are in ascending or descending order.

Solution:

Let $f(n)$ be the minimum number of moves required to 'arrange' any permutation of the n cards. Suppose we have a permutation with starting card k. In $f(n-1)$ moves we can arrange the remaining $(n-1)$ cards to get either the sequence $(k, 1, 2, \ldots, k-1, k+1, \ldots, n)$ or $(k, n, n-1, \ldots, k+1, k-1, \ldots, 2, 1)$. In one move, we can make the former sequence $(k, k-1, k-2, \ldots, 1, k+1, k+2, \ldots, n)$ and with one more move we get the sequence $(1, 2, 3, \ldots, n)$ and we are done. Similarly in the latter case we need only two additional moves to get $(n, n-1, \ldots, 1)$. Thus, in either case, we can complete the task using $f(n-1)+2$ moves, so $f(n) \leq f(n-1) + 2$.

Now, to prove the bound for general $n \geq 5$, it suffices to prove it for $n = 5$ and then induct using $f(n) \leq f(n-1) + 2$. The proof of $f(5) = 4$ we left to the reader as an exercise.

Example 2.2.2 (IMO 2017)
An integer $N \geq 2$ is given. A collection of $N(N+1)$ soccer players, no two of whom are of the same height, stand in a row. Sir Alex wants to remove $N(N-1)$ players from this row leaving a new row of $2N$ players in which the following N conditions hold:

1. *no one stands between the two tallest players*
2. *no one stands between the third and fourth tallest players*

   $\dots$

N. *no one stands between the two shortest players.*

Show that this is always possible.

Solution:

We will crack this problem by induction on N with an algorithm.

Base: The case when $n = 1$ is trivial; the original two soccer players work.

Induction hypothesis: Sir Alex has an algorithm for $N(N-1)$ soccer players.

Induction step: Split the row of players into N segments, each with $N+1$ people. Consider the $N+1$ tallest people, at least two of them will be in one of the segments by Pigeonhole principle. WLOG, let it be the first segment. Call these two people Timur and Amir.

Then we can remove all the other players in the first segment (apart from Amir and Timur), which gives $N-1$ players deleted. Next, we remove the rest of the $N+1$ tallest players (again, excluding Timur and Amir), which gives at most another $N-1$ players deleted. Hence, at most $2(N-1)$ players were removed and we still have at least $N(N-1)$ people. (excluding Timur and Amir), so we can apply the induction hypothesis to get $2(N-1)$ players, and we can get the desired result. Lastly, since Amir and Timur will now be taller than these $2(N-1)$ players (and they will be next to each other), we can add them in and all the conditions still hold. This completes the induction.

2.3 Invariants and Monovariants

Here, we will introduce another crucial idea in algorithm problems: invariants and monovariants. To reiterate from previous chapter: invariant is a quantity that does not change, and monovariant is a quantity that changes monotonically: it is either always increasing or decreasing. We will start with a simple, introductory example:

Example 2.3.1
Some numbers are written in the cells of the table $m \times n$. It is allowed to change the sign of all numbers in one column or in one row at the same time. Prove that by several such operations it is possible to ensure that the sums of the numbers in any row and in any column are nonnegative.

Solution:

Consider the following algorithm: if the sum of the numbers in a row (or in a column) is negative, then we will change the sign of the numbers in this row (column). To prove that this process stops, we will find some characteristics of the table, which increases monotonously with each step. The desired characteristic is the sum of all cells. This amount increases at each step. The process will end, since the number of character arrangements for numbers is finite.

Example 2.3.2 (Canada 2014)
A number of robots are placed on the squares of a finite, rectangular grid of squares. A square can hold any number of robots. Every edge of each square of the grid is classified as either passable or impassable. All edges on the boundary of the grid are impassable. You can give any of the commands up, down, left, or right.
ll of the robots then simultaneously try to move in the specified direction. If the edge adjacent to a robot in that direction is passable, the robot moves across the edge and into the next square. Otherwise, the robot remains on its current square. You can then give another command of up, down, left, or right, then another, for as long as you want. Suppose that for any individual robot, and any square on the grid, there is a finite sequence of commands that will move that robot to that square. Prove that you can also give a finite sequence of commands such that all of the robots end up on the same square at the same time.

Solution:

Clearly, it suffices to consider the case where two robots are on the grid. Define the distance between two robots to be the minimum number of commands needed for a robot to travel from one of the squares to the other.

Let A and B be the two robots on the grid. We will provide an algorithm that, when repeatedly invoked, eventually decreases the distance between A and B. In particular, the algorithm is to give the commands that would move A to B's position with the minimum distance. This works since A will not hit a wall (otherwise there is a "faster" way to move A), but since the rectangular grid is finite, B will eventually hit a wall after a few runs of this algorithm. Therefore, the distance between A and B after running the algorithm eventually decreases, and we are done.

Example 2.3.3 (Russia 2009)
Around a circle there are 2009 integer nonnegative numbers, each at most 100. You are allowed to add 1 to two neighboring numbers, and any given pair of neighbors may be operated on at most k times. For which minimal k is it guaranteed that one can make all the numbers equal?

Solution:

Let the numbers on the circle be $a_1, a_2, \ldots, a_{2009}$, and extend them periodically by $a_{n+2009} = a_n$. Put $N = 100400$.

For the lower bound take $a_2 = a_4 = \cdots = a_{2008} = 100$ and $a_1 = a_3 = \cdots = a_{2009} = 0$.

Consider the sum $S = (a_2 - a_3) + (a_4 - a_5) + \cdots + (a_{2008} - a_{2009})$.

This sum increases by 1 when we add 1 to the pair (a_1, a_2), decreases by 1 when we add 1 to the pair (a_{2009}, a_1), and does not change for any other operation. The initial value equals $S_0 = 100 \cdot 1004 = N$, while the final value (when all numbers are equal) must be 0. Hence the pair (a_{2009}, a_1) must have been increased at least N times, so $k \geq N$.

For sufficiency let $a_1, \ldots, a_{2009}$ be arbitrary. For each i increase the pair (a_i, a_{i+1}) exactly $s_i = a_{i+2} + a_{i+4} + \cdots + a_{i+2008}$ times. Then each a_i becomes

$$a_i + s_{i-1} + s_i = a_i + (a_{i+1} + a_{i+3} + \cdots + a_{i+2007}) + (a_{i+2} + a_{i+4} + \cdots + a_{i+2008}) = a_1 + \cdots + a_{2009}$$

so all numbers become equal. Finally, $s_i \leq 1004 \cdot 100 = N$, therefore $k = N$ suffices.

Example 2.3.4 (APMO 1997)
n people are seated in a circle. A total of nk coins have been distributed among them, but not necessarily equally. A move is the transfer of a single coin between two adjacent people. Find an algorithm for making the minimum possible number of moves which result in everyone ending up with the same number of coins.

Solution:

We want each person to end up with k coins. Let the people be labeled from $1, 2, \ldots, n$ in order (note that n is next to 1 since they are sitting in a circle). Suppose that person i has c_i coins. Now, we introduce the variable $d_i = c_i - k$, since this indicates how close a person is to having the desired number of coins. Consider the quantity $X = |d_1|+|d_1+d_2|+|d1+d_2+d_3|+\cdots+|d_1+d_2+\cdots+d_{n-1}|$.

Clearly $X = 0$ if and only if everyone has k coins, so our goal is to make $X = 0$. The reason for this choice of X is that moving a coin between person j and person $j+1$ for $1 \leq j \leq n-1$ changes X by exactly 1 as only the term $|d_1 + d_2 + \cdots + d_j|$ will be affected. Hence, X is a monovariant and is fairly easy to control (except when moving a coin from 1 to n or vice versa). Let $s_j = d_1 + d_2 + \cdots + d_j$.

We claim that as long as $X > 0$ it is always possible to reduce X by 1 by a move between j and $j+1$ for some $1 \leq j \leq n-1$.

We use the following algorithm. Assume WLOG $d_1 \geq 1$. Take the first j such that $d_{j+1} < 0$. If $s_j > 0$, then simply make a transfer from j to $j+1$. This reduces X by one since it reduces the term $|s_j|$ by one.

The other possibility is $s_j = 0$, which means $d_1 = d_2 = \cdots = d_j = 0$ (recall that d_{j+1} is the first negative term). In this case, take the first m > i+1 such that $d_m \geq 0$. Then, $d_{m-1} < 0$ by the assumption on m, so we move a coin from m to $(m-1)$.

Note that all terms before dm were either 0 or less than 0 and $d_{m-1} < 0$, so s_{m-1} was less than 0. Our move has increased sm-1 by one, and has hence decreased $|s_{m-1}|$ by one, so we have decreased X by one.

Thus, at any stage, we can always decrease X by at least one by moving between j and $j+1$ for some $1 \leq j \leq n-1$. We have not yet considered the effect of a move between 1 and n. Thus our full algorithm is as follows: At any point of time, if we can decrease X by moving a coin from 1 to n or n to 1, do this. Otherwise, decrease X by 1 by the algorithm described in the above paragraph.

2.4 Information

In previous examples, we encountered problems that required us just to prove something or bound something by reference to smaller cases, like in the case of induction or recursion, but in this section we will get acquainted with other methods of bounding, primarily by using common sense, and methods of constructing algorithms.

Let's start with easy one:

Example 2.4.1
You have 9 coins that look the same, one of them is fake, and weights less than real ones. Can you find the fake coin in 2 weightings on a balance scale.

Before beginning the solution we need to understand that in this types of questions it only makes sense to compare the equal number of coins at one time, for example 3 coins vs 3 coins.
We do not know the actual weights of the real and fake coins, so if we were to compare non-equal number of coins with each other, we can manipulate the numerical values of weights to get every possible outcome.

Answer: $\boxed{1}$.

What is the best way of solving easy problem – make even easier problem!

Consider that you have 3 coins and need to find a fake one in just one comparison.

Do the only thing you can do – compare 1st and 2nd coins, if either of them is lighter than the other, then it is the fake one, if they weight the same, then fake one must be the last, 3rd coin.

Now, back to the original question: first, split coins in 3 groups of 3 coins each, and by treating each group as individual big coin use previous problem to find a group with a fake one in just one move, spend the second move to find the fake one in the group, again, using the previous problem.

Let's solve the harder version of this problem:

Example 2.4.2
You have n coins that look the same, one of them is fake, and weights less than real ones. What is the minimal number of comparisons to find the fake coin?

Answer: $\boxed{1}$.

To proof that such number of moves is enough, repeatedly split coins into 3 groups of roughly the same size and compare two of them (transfer 1 coin if needed).

Now, to the bound – say that we need at least m comparisons to find the coin.

Every possible strategy of finding fake coin in a fixed number of comparisons can be written like this:

- (a_1-th + a_2-th + ... + a_k-th) vs (b_1-th + b_2-th + ... + ab_k-th)
  - if we get ">" the compare ...
  - if we get "<" the compare ...
  - if we get "=" the compare ...

After m comparisons we have a sequence of $>$, $<$, $=$ and from that we need to conclude which coins is the fake one.

The total number of sequences that can be formed is 3^m, and number of coins is n.

Since every coin can be fake – that is you can swap the any two coins and still be able to find the fake one – one we must have $3^m \geq n$, so $k \geq \lceil \log_3(n) \rceil$.

Example 2.4.3 (Russia 2005)
Different numbers are written on the backs of the 2005 cards (one on each one). In one question, you can point to any three cards and find out the set of numbers written on them. What is the minimum number of questions to find out which numbers are written on each card?

Solution:

Okay, problem statement is clear, but how can we approach bounding? In such situations, we will examine the minor facts or observations that might seem obvious and useless. First of all, it is clear that each card is involved in at least one inquiry, otherwise we will not determine the number on it. Assume that we asked N questions. Let there be k cards that participated in exactly one question. Then, there cannot be two such cards in one question. Indeed, if two of such cards participated in the same question, then swapping the numbers on these cards would not change the answers to the questions; therefore, it is impossible to determine which number is written on which of them. Therefore, $k \leq N$. The rest of the cards participated in at least two questions. Also, if we sum up the number the number of questions for each card in which it participated, we get a tripled number of questions.

Therefore, $3N \geq k + 2(2005 - k) = 4010 - k \geq 4010 - N$, so, $2N \geq 2005 \to N \geq 1003$.

Now we will present an algorithm to find the numbers on the cards for 1003 questions. Let's put aside one card, and divide the rest into 334 groups of 6 cards. In each group, we will number the cards with numbers from 1 to 6 and ask three questions: (1,2,3) , (3,4,5) and (5,6,1) .

Then the numbers on cards 1, 3, and 5 occur in two answers (for different cards, in different pairs), and they are uniquely determined. Also, the numbers on cards 2, 4, and 6 are the remaining numbers in each of the answers. So for $\frac{2004}{6} * 3 = 1002$ questions, we will find the numbers on the 2004 cards.

It remains to ask about the deferred card along with any two already known ones.

Example 2.4.4 (Russia 1997)
Magician and magician helper perform a magic trick as follows. A spectator writes on a board a sequence of N (decimal) digits. Magician helper covers two adjacent digits with a black disc. Then, a magician comes and says both closed digits (and their order). For which minimal N can this trick always work?

Solution:

Suppose that we can perform such a trick on a certain N. If so, a magician can uniquely restore the original sequence of digits from each variant that can be written by the spectator.

In other words, the number of variants that can be written on the board should be less than number of sequences with 2 hidden digits. Magician helper may have in front of magician $(N-1)10^{N-2}$ different configurations ($N-1$ positions for the disc and 10^{N-2} for the reminder), and spectator can choose 10^N configurations. So $(N-1)10^{N-2} \geq 10^N$ and $N \geq 101$

Strategy for $n = 101$:

Let the sum of all the digits in the odd positions have a remainder s divided by 10, and the sum of all the digits in the even positions have a remainder t divided by 10 (the positions are numbered from left to right by numbers from 0 to 100). Let's assume $p = 10s + t$. At this moment, assistant closes the numbers at positions p and $p+1$. After seeing which numbers are closed, the magician will determine p, and therefore s and t. Note that one closed digit is in an odd position, and the other is in an even position. Thus, by calculating the sum of the open digits in the odd positions and knowing s, the magician will determine the closed digit in the odd position. Similarly, a closed digit is determined in an even position.

Example 2.4.5 (IMSC 2025)
*There are $3^n + 1$ pebbles of 11 grams in a row. Two of them are **fat** and weigh 12 grams, and they are neighbors. In one query, it is possible to ask the weight of any set of pebbles. Find the minimum number of queries to determine the fat pair regardless of the answers to the queries. In addition, queries must be determined in advance. In other words, it is not possible to change queries depending on the answers for them.*

Solution:

In general, there are 3^n possible positions for our **fat** pair. In addition, every inquiry gives us information about the number of fat stones in the selected set. Assume that there are k inquires, so there is maximum 3^k cases, so $3^k \geq 3^n \to k \geq n$.

2.5 Crazy algorithms

Example 2.5.1 (JetBrains 2024)
In the vertices of a polygon $A_1, A_2, \ldots, A_{100}$, there are initially 30 zeros and 70 ones. There are 100 tokens numbered from 1 to 100. In one move, it is allowed to perform one of the following actions:

1. *If token i has not been used yet, then it can be placed on vertex A_i.*
2. *If there is a token on vertex A_j, then it is possible to change the number in this vertex (from 0 to 1 or from 1 to 0). If the number changes from 0 to 1, the token is moved to vertex A_{j+1} (where $A_{101} = A_1$); if the number changes from 1 to 0, the token is discarded completely.*

At no point may two tokens occupy the same vertex. After several moves, all 100 tokens have been discarded. How many zeros can be present in the vertices at this moment?

Answer: $\boxed{1}$.

Redefine the problem statement:

We have 3×100 board with the top row filled with tokens and initially 30 zeros and 70 ones on the second row. On any move tokens changes the number of the cell (from 0 to 1 or from 1 to 0). If the number changes from 0 to 1, the token is moved to the right, if the number changes from 1 to 0, the token moves down. After some number of moves every token is now on the third row. How many zeros can be present in the second row at this moment?
Note: we allow for the collisions between tokens.

1	1	1	1	1	1	$\cdots$	1	1
0	0	1	0	1	1	$\cdots$	0	0
1	1	1	1	1	1	$\cdots$	1	1

We assign the value to each token depending on the cell token is in as follows:

100	99	98	97	96	95	$\cdots$	2	1
101	100	99	98	97	96	$\cdots$	3	2
102	101	100	99	98	97	$\cdots$	4	3

If we count the total sum of values of tokens and values in the cells of the table we will get

$$100 + 70 + 100 + (100 + 99 + 98 + \cdots + 2 + 1) = 5320.$$

Note that on any move of the token total sum remains constant $\pmod{100}$.

> Claim: at the end, no 2 tokens lie in the same cell.

So if we prove the claim the total sum at the end will be

$$0 + \# \text{ of ones in the 2nd row} + 100 + (102 + 101 + 100 + \cdots + 4 + 3) =$$

$$5350 + \# \text{ of ones in the 2nd row} \equiv 5320 \pmod{100}$$

So the # of ones in the 2nd row will be 70.

FTSoC assume that at some point 2 tokens lie in the same cell on the 3rd row. Now observe that both of them need to come in that space by moving downwards and changing the number on the 2nd row from 1 to 0. Between them we must have a token to change the number from 0 to 1, so at least 3 tokens visited the cell on the 2nd row.

Now look at the first time 3 tokens visited some cell (call it A, and the cell on the left of it – B), only one of them could have come from above, so at least 2 of them came from B. But by the same logic, for 2 tokens to move right, we need 1 token to move down, so at least 3 tokens visited B – contradiction.

> **Example 2.5.2 (Kazakhstan 2023)**
> *Let G be a graph whose vertices are 2000 points in the plane, no three of which are collinear, that are colored in red and blue and there are exactly 1000 red points and 1000 blue points. Given that there exist 100 red points that form a convex polygon with every other point of G lying inside of it. Prove that one can connect some points of the same color such that segments connecting vertices of different colors do not intersect, and one can move from a vertex to any vertex of the same color using these segments.*

Solution:

The problem might seem as tough, and indeed it is tough. However, the problem statement gives an exact clue to construct our algorithm.

Let's call a triangle $2B$ if it has two vertices in black (connected by an edge) and one in red. It is similar to $2R$ if two vertices are red (connected by an edge) and one is black.

Let's define the operation $\oplus$ on a triangle $2B$ or $2R$ as follows:

- WLOG consider a $2R$ triangle.
- If there are no black dots inside this triangle, then connect all the red dots inside the triangle with one of the red vertices of $2R$ and finish the operation. Please note that the new red dots will be connected to at least one previous red dot (to connect the red dots).
- If there is at least one black dot inside $2R$, then take any of them and connect it to the black vertex of $2R$. Note that this black dot is connected to the previous black vertex of the triangle.2R (for connecting black dots). Then we will divide the triangle into three parts as shown in the figure above or belowThus, 2R is divided into two 2B and one 2R, and then we make $\oplus$ for each of the resulting parts.
- Since any triangle $2R$ or $2B$ contains only a finite number of points inside, the operation will end at some point.

Let's take the original graph and connect all neighboring vertices of the convex hull with an edge. Then we take any black point inside a convex 100-gon and divide the graph into 100 triangles of the form 2R, and perform $\oplus$ for each of them. From the construction, it can be understood that the newly found points inside the triangles have an edge with at least one previous vertex of the same color. From this we can conclude that the red and black dots are connected. In addition, no two multicolored edges intersect because we have performed a triangulation that does not allow for intersection.

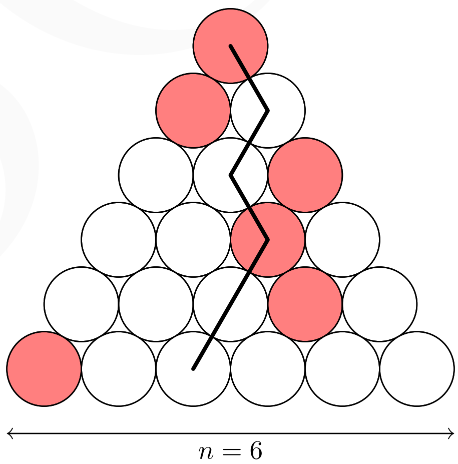
No segments (two segments can intersect only if they have a common vertex, but then all the ends of these two segments are the same color).

2.6 Problems

Problem 2.6.1 (APMO 2022) Let n and k be positive integers. Cathy is playing the following game. There are n marbles and k boxes, with the marbles labeled 1 to n. Initially, all marbles are placed inside one box. Each turn, Cathy chooses a box and then moves the marbles with the smallest label, say i, to either any empty box or the box containing marble $i+1$. Cathy wins if at any point there is a box containing only marble n. Determine all pairs of integers (n, k) such that Cathy can win this game.

Problem 2.6.2 (All-Russian 2014) There are n cities in the state, and an express runs between each two of them (in both directions). For any express train, the prices of round-trip and round-trip tickets are equal, and for any different express trains these prices are different. Prove that a traveler can choose the starting city, leave it and travel sequentially on $n-1$ express trains, paying less for the fare on each next one than for the fare on the previous one. (A traveler may enter the same city several times.)

Problem 2.6.3 (IMO 2023) Let n be a positive integer. A Japanese triangle consists of $1+2+\cdots+n$ circles arranged in an equilateral triangular shape such that for each $i = 1, 2, \ldots, n$, the i-th row contains exactly i circles, exactly one of which is colored red. A ninja path in a Japanese triangle is a sequence of n circles obtained by starting in the top row, then repeatedly going from a circle to one of the two circles immediately below it and finishing in the bottom row. Here is an example of a Japanese triangle with $n = 6$, along with a ninja path in that triangle containing two red circles.



In terms of n, find the greatest k such that in each Japanese triangle there is a ninja path containing at least k red circles.

Problem 2.6.4 (IMO 2020) There are $4n$ pebbles of weights $1, 2, 3, \ldots, 4n$. Each pebble is colored in one of n colors and there are four pebbles of each color. Show that we can arrange the pebbles into two piles so that the following two conditions are both satisfied:

- The total weights of both piles are the same.
- Each pile contains two pebbles of each color.

Problem 2.6.5 (All-Russian 2005) 100 people from 25 countries, four from each country, sit in a circle. Prove that one may partition them onto 4 groups in such way that no two countrymen, nor two neighboring people in the circle, are in the same group.

Problem 2.6.6 (IMO 2010) Each of the six boxes B_1, B_2, B_3, B_4, B_5, B_6 initially contains one coin. The following operations are allowed

- Choose a non-empty box B_j, $1 \leq j \leq 5$, remove one coin from B_j and add two coins to B_{j+1}
- Choose a non-empty box B_k, $1 \leq k \leq 4$, remove one coin from B_k and swap the contents (maybe empty) of the boxes B_{k+1} and B_{k+2}.

Determine if there exists a finite sequence of operations of the allowed types, such that the five boxes B_1, B_2, B_3, B_4, B_5 become empty, while box B_6 contains exactly $2010^{2010^{2010}}$ coins.

Problem 2.6.7 You have n coins, one of them is fake one, but you do not know if it is lighter or heavier than the real coins. Can you find the fake coin and determine if it is lighter or heavier than the real one if:

a) $n = 12$

b) $n = 14$

c) $n = 13$

Problem 2.6.8 (ISL 2024 C3) Let n be a positive integer. There are $2n$ knights sitting at a round table. They consist of n pairs of partners, each pair of which wishes to shake hands. A pair can shake hands only when next to each other. Every minute, one pair of adjacent knights swaps places. Find the minimum number of exchanges of adjacent knights such that, regardless of the initial arrangement, every knight can meet her partner and shake hands at some time.

Problem 2.6.9 (USAJMO 2020) Let $n \geq 2$ be an integer. Carl has n books arranged on a bookshelf. Each book has a height and a width. No two books have the same height, and no two books have the same width. Initially, the books are arranged in increasing order of height from left to right. In a move, Carl picks any two adjacent books where the left book is wider and shorter than the right book, and swaps their locations. Carl does this repeatedly until no further moves are possible. Prove that regardless of how Carl makes his moves, he must stop after a finite number of moves, and when he does stop, the books are sorted in increasing order of width from left to right.

Problem 2.6.10 (IMO 2024) Turbo the snail plays a game on a board with 2024 rows and 2023 columns. There are hidden monsters in 2022 of the cells. Initially, Turbo does not know where any of the monsters are, but he knows that there is exactly one monster in each row except the first row and the last row, and that each column contains at most one monster.

Turbo makes a series of attempts to go from the first row to the last row. On each attempt, he chooses to start on any cell in the first row, then repeatedly moves to an adjacent cell sharing a common side. (He is allowed to return to a previously visited cell.) If he reaches a cell with a monster, his attempt ends and he is transported back to the first row to start a new attempt. The monsters do not move, and Turbo remembers whether or not each cell he has visited contains a monster. If he reaches any cell in the last row, his attempt ends and the game is over.

Determine the minimum value of n for which Turbo has a strategy that guarantees reaching the last row on the n-th attempt or earlier, regardless of the locations of the monsters.

Problem 2.6.11 (Kazakhstan 2025, Regional) A and B are playing a game on a 100×100 checkered board. Each player has a chip. At the beginning of the game, player A's chip is in the lower-left corner, and player B's chip is in the lower-right corner. The players take turns making moves, starting with A. In one turn, the player moves his chip to any square of the board adjacent to the square of the previous position. Prove that player A can achieve in a finite number of moves that at some point his chip will be on the same square as player B chip, regardless of the moves of the second player.

Problem 2.6.12 (All-Russian 2015) The field is a 41×41 checkered square, in one of the cells of which a tank is disguised. The fighter fires at one cell in one shot. If a hit occurs, the tank crawls over to the cell next to the side of the field, if not, it remains in the same cell. At the same time, after the shot, the fighter pilot does not know if a hit has occurred. To destroy a tank, you need to hit it twice. What is the least number of shots you can do to ensure that the tank is destroyed?

Problem 2.6.13 (Saint-Petersburg 1994) The pluses and minuses are placed in the cells of the 1995×1995 table. It is allowed to select 1995 cells, no two of which are in the same row or column, and change the signs in the selected cells. Prove that using such operations it is possible to ensure that no more than 1994 pluses remain in the table.

Problem 2.6.14 (Saint-Petersburg 2018) In a 9×9 table, all cells contain zeros. The following operations can be performed on the table:

1. Choose an arbitrary row, add one to all the numbers in that row, and shift all these numbers one cell to the right (and place the last number in the first position).

2. Choose an arbitrary column, subtract one from all its numbers, and shift all these numbers one cell down (and place the bottommost number in the top cell).

Is it possible to obtain a table in which all cells, except two, contain zeros, with 1 in the bottom-left cell and -1 in the top-right cell after several such operations?

Problem 2.6.15 (USAMO 2011) An integer is assigned to each vertex of a regular pentagon so that the sum of the five integers is 2011. A turn of a solitaire game consists of subtracting an integer m from each of the integers at two neighboring vertices and adding $2m$ to the opposite vertex, which is not adjacent to either of the first two vertices. (The amount m and the vertices chosen can vary from turn to turn.) The game is won at a certain vertex if, after some number of turns, that vertex has the number 2011 and the other four vertices have the number 0. Prove that for any choice of the initial integers, there is exactly one vertex at which the game can be won.

Problem 2.6.16 (239 olympiad 2025) The numbers from 1 to 2025 are arranged in some order in the cells of the 1×2025 strip. Let's call a flip an operation that takes two arbitrary cells of a strip and swaps the numbers written in them, but only if the larger of these numbers is located to the left of the smaller one. A flop is a set of several flips that do not contain common cells that are executed simultaneously. (For example, a simultaneous flip between the

2nd and 8th cells and a flip between the 5th and 101st cells.) Prove that there exists a sequence of 66 flops such that for any initial arrangement, applying this sequence of flops to it will result in the numbers being ordered from left to right in ascending order.

Problem 2.6.17 (Russia 2004) On the table there are 2004 boxes, each containing one ball. It is known that some of the balls are white, and their total number is even. You are allowed to point to any two boxes and ask whether there is at least one white ball among them. What is the minimum number of questions that is sufficient to guarantee identifying at least one box that contains a white ball?

Problem 2.6.18 (Russia 2018) In a card game, each card is assigned a numerical value from 1 to 100, where each card beats a smaller one, with one exception: card 1 beats card 100. The player knows that there are 100 cards with distinct values placed face down in front of him. The dealer, who knows the exact order of these cards, can, for any chosen pair of cards, inform the player which one beats the other.

Prove that the dealer can make one hundred such announcements, after which the player will be able to determine the exact value of every card.

Problem 2.6.19 (Russia 2019) There are n coins of pairwise distinct weights and n balance scales, where $n > 2$. In each weighing, you may choose any one of the scales, place one coin on each of its pans, observe the result, and then remove the coins. Some of the scales (unknown which) are faulty and may randomly show either the correct or an incorrect result. What is the smallest number of weighings that guarantees finding the heaviest coin?

Chapter 3

Processes

In this chapter, we will consider processes. It might seem that processes and algorithms are similar topics, but the main difference is that we should construct something by ourselves, whereas in processes we can not create something new in problem, instead we can make judgments on which we have.

3.1 Invariants and monovariants

As always, we will start with invariants, and monovariants. Fortunately, the idea of invariants is a very useful approach to practically every process problem: it might be very helpful to make several observations on processes.

Example 3.1.1 (ISL 2014)
We have 2^m sheets of paper, with the number 1 written on each of them. We perform the following operation. In every step we choose two distinct sheets; if the numbers on the two sheets are a and b, then we erase these numbers and write the number $a+b$ on both sheets. Prove that after $m2^{m-1}$ steps, the sum of the numbers on all the sheets is at least 4^m .

Solution:

Ok, this problem is felt like it can be fully cracked by invariants, so let us try to find this. The first thought that comes to mind is to consider the sum of numbers written on cards, but it is a wrong approach; rather, we should consider the product of the 2^m numbers, instead of the sum.

Claim: Whenever the operation is performed, the resulting product of the 2^m increases by at least 4 times.

Proof: $(a+b)^2 \geq 4ab$.

Now, at the end of the $m2^{m-1}$ steps, the product will be at least $4^{m2^{m-1}} = 2^{2^m}$, so by AM-GM, the sum of the 2^m sheets of paper is at least $2^m \cdot 2^m = 4^m$, as desired.

Example 3.1.2 (IMO 1986)
To each vertex of a regular pentagon an integer is assigned, so that the sum of all five numbers is positive. If three consecutive vertices are assigned the numbers x, y, z respectively, and $y < 0$, then the following operation is allowed: x, y, z are replaced by $x+y, -y, z+y$ respectively. This iteration is performed repeatedly as long as at least one of the five numbers is negative. Determine whether this procedure necessarily comes to an end after a finite number of steps.

Solution:

Answer: This procedure is always terminates.

First, if all our numbers on the pentagon are labeled x_i, all the indices have taken the undo module 5. Let in the first step $(x_1, x_2, x_3, x_4, x_5) \to (x_1, x_2 + x_3, -x_3, x_4 + x_3, x_5)$

We will solve this problem using monovariants. It is not hard to see that sum of all numbers written on the pentagon remains invariant. Because the sum is positive and remains invariant and at the end all numbers are also positive, this leads to the thought that pairwise differences should be small, so it can be an idea to consider pairwise differences. After several trials we can end with such monovariant:

$$f_0(x_1, x_2, x_3, x_4, x_5) =$$
$$\frac{1}{2}\sum_{i=1}^{5}(x_{i+1} - x_i)^2.$$

WlOG assume that $x_4 < 0$, and after meticulous opening brackets it is not hard to see that our quantity is decreasing step by step. Because: $f_{\text{new}} - f_{\text{old}} = Sx_4 < 0$, where S is sum of all numbers in pentagon. Because our quantity is decreasing, the process terminates after finite number of steps.

In previous examples we have seen the problems related to monovariant, and of course we can solve certain algorithmical problems with invariants,

Example 3.1.3 (All-Russian 1997)
There are some stones placed on an infinite (in both directions) row of squares labeled by integers. (There can be more than one stone on a given square). There are two types of moves:

- *Remove one stone from each of the squares n and $n-1$ and place one stone on $n+1$*
- *Remove two stones from square n and place one stone on each of the squares $n+1$ and $n-2$.*

Show that at some point no more moves can be made.

Solution:

Give a stone in square k weight φ^k where $\varphi = \frac{1+\sqrt{5}}{2}$, and it is not hard to see that the sum of all weights remains invariant due to the relation $\varphi^k = \varphi^{k-1} + \varphi^{k-2}$. Suppose that the process never terminates, from which there exists at least one square that is operated on infinitely many times. Consider the square of least value that is operated on finitely many times, and note that the square to the left of it is operated on infinitely, so arbitrarily many stones build up on the former square, so it must be operated on infinitely, contradiction. Thus all squares are operated on infinitely, so the index of the rightmost square that contains a stone grows arbitrarily large. At some point, the weight of the stone in that square exceeds the initial total weight, contradicting the invariant.

Example 3.1.4

3.2 Induction

Example 3.2.1

Although not only direct induction exists, you might encounter problems that use nested induction or double induction. The next example involves this idea.

Example 3.2.2 (EGMO 2018)
The n contestant of EGMO are named $C_1, C_2, \cdots C_n$. After the competition, they queue in front of the restaurant according to the following rules.

1. *The Jury chooses the initial order of the contestants in the queue.*
2. *Every minute, the Jury chooses an integer i with $1 \leq i \leq n$. If contestant C_i has at least i other contestants in front of her, she pays one euro to the Jury and moves forward in the queue by exactly i positions.*
3. *If contestant C_i has fewer than i other contestants in front of her, the restaurant opens and process ends.*

Prove that the process cannot continue indefinitely, regardless of the Jury's choices.

Solution:

Starting from the position farthest from the restaurant, label the positions $P_1, P_2, \ldots, P_n$ (i.e. the contestant at position P_n is closest to the restaurant).

First induct on n. The result is obvious when $n = 1, 2$. Therefore, assume that the result is true for $n-1$. Suppose that initially contestant C_n is at the position P_m. The main observation is that the Jury never can choose C_n. We claim that C_n must reach P_1 sooner or later. To show this we again induct, this time on m. For $m = 1$, the claim is true (C_n is at P_1 initially only!!). Suppose our claim is true for $m-1$.

Now, let A denote the set of students placed at $P_1, P_2, \ldots, P_{m-1}$; and B denote the set of students placed at $P_{m+1}, P_{m+2}, \ldots, P_n$. If at any moment of time, some contestant from A jumps over to B (i.e. crosses C_n), then C_n will move to position P_{m-1}, and we can apply the second inductive hypothesis to conclude that C_n must reach P_1. Once, C_n reaches P_1, we can effectively ignore its presence (as it can never be the cause of any contestant's jump). Then using the first inductive hypothesis, we get that the game must terminate.

Otherwise no contestant from the group A ever crosses over to B. Then these two group of contestants will never interact with each other, and so the presence of C_n is superfluous (once again it does not cause anyone's move). Hence, we can ignore C_n, and apply the first inductive hypothesis on the remaining $n-1$ contestants to get the desired conclusion. Hence, done.

3.3 Chip firing

Here we will introduce with a tough topic from "adult" mathematics. Despite this fact, many problem proposers assumed that it is a good idea to give such problems on school Olympiads. As usual, we will start with easy, demonstrative example.

Example 3.3.1 (ISL 1994)
1994 *girls are seated in a circle. Initially one girl is given* n *coins. In one move, each girl with at least 2 coins passes one coin to each of her two neighbors.*

a Show that if $n < 1994$*, the game must terminate.*

b Show that if $n = 1994$*, the game cannot terminate.*

Solution:

Solution for part a): Okay, the problem statement looks very familiar, for example at APMO 1997 we have faced with similar problem. As usual let's label the girls $G_1, G_2, \ldots, G_{1994}$ all indices under modulo 1994.

Suppose that the game does not terminate. Then some girl must pass coins infinitely times. If some girl passes only finitely many times, there exist two adjacent girls, one of whom has passed finitely many times and one of whom has passed infinitely many times. The girl who has passed finitely many times will then accumulate coins indefinitely after her final pass, which is impossible. Hence,every girl must pass coins infinitely many times.

Now the key idea is the following: For any two neighboring girls G_i and G_{i+1}, let c_i be the first coin ever passed between them. After this, we may assume that c_i always stays stuck between G_i and G_{i+1}, because whenever one of them has c_i and makes a move, we can assume that the coin passed to the other girl was c_i. Therefore, each coin is eventually stuck between two girls. Since there are fewer than 1994 coins, this means that there exist two adjacent girls who have never passed a coin to each other. This contradicts the result of the first paragraph.

Solution for part b) This is simple using invariants. Let a coin with girl i have weight i, and let G_1 initially have all the coins. In each pass from $G_i$to her neighbors, the total weight either does not change or changes by ± 1994 (if G_1 passes to G_{1994} or vice versa). So, the total weight is invariant $\pmod{1994}$. The initial weight is 1994, so the weight will always be divisible by 1994. If the game terminates, then each girl has one coin, so the final weight is $1+2+3+\cdots+1994 = \frac{1994\cdot 1995}{2}$ which is not divisible by 1994 – contradiction.

For the experienced combinatorics solver the problem might seem cropped. Definitely, there exists generalization of this problem: what if graph is not a mere cycle, but an arbitrary, connected graph? Or what if the graph is disconnected? Now, we will provide a generalized version of problem:

Example 3.3.2
Given a connected graph G with n nodes and m edges (without loops or double edges), and N chips. Start by putting a_i chips on node i, $i = 1, 2, \ldots, n$. Such that $a \in \mathbb{Z}_0$ and $\sum_{i=1}^{n} a_i = N$. Recall that firing the node i means that we decrease a_i by the $\deg(i)$, and for each neighbor j of i increase a_j by 1. Which chip firing games are infinite and which are finite?

Solution:

a) If $N > 2m - n$ then the game is infinite.

b) If $m \le N \le 2m - n$ then there exists an initial configuration guaranteeing finite termination and also one guaranteeing infinite game.

c) If $N < m$ then the game is finite.

It is obvious that if $N > 2m - n$ then the game cannot terminate: there is always a node v with at least $\deg(v)$ chips on it. It is also obvious that if $N \le 2m - n$ then we can place at most $\deg(v) - 1$ chips on each node v, and so there are configurations with no legal move.

Next we show that if the number of chips is at least m then there is an initial configuration that leads to an infinite game. Clearly it suffices to show this for $N = m$. Consider any acyclic orientation of G, and let $\deg^+(v)$ denote the out-degree of node v. Let us place $\deg^+(v)$ chips on each node v; this is clearly possible since there are m chips altogether. We claim that this game is infinite.

Observe first that there must exist a node that can be fired in the first step. In fact, the orientation is acyclic, which implies that there is a source, i.e., a node v with $\deg(v) = \deg^+(v)$. Now fire this node, and observe that the resulting distribution of chips can also be obtained from an orientation: if we reverse the edges incident with v, we decrease the outdegree of v by $\deg^+(v)$, and increase the outdegree of each of its neighbors by 1. Since reversing the edges incident with a source does not create any directed cycle, we can find a source in the resulting digraph which can again be fired, etc.

The proof of (c) is motivated by the previous construction. Consider any distribution of $N < m$ chips on the nodes; let $f(v)$ denote the number of chips on node v. Also consider an acyclic orientation of the graph G and the quantity

$$T = \sum_{v \in V(G)} \max\{0, f(v) - \deg^+(v)\}.$$

We say that a node u is *deficient* if $f(u) < \deg^+(u)$; by our hypothesis that $N < m$, there must exist a deficient node. We are going to show that we can modify the orientation during the game so that T never increases and if the set of deficient nodes changes then T must actually decrease. If the game is infinite then every node gets fired infinitely often, and hence the set of deficient nodes must change infinitely often (since a deficient node cannot be fired). Since T cannot decrease infinitely often, this implies that the game is finite.

Consider the node v that is first fired; we have $f(v) \geq \deg(v)$. Fire v and reverse the orientation of all edges leaving v. We do not create any cycle. Moreover, we do not increase T since the term in T corresponding to v decreases by $\deg(v) - \deg^+(v)$ while each of the $\deg(v) - \deg^+(v)$ terms corresponding to the nodes u for which $uv \in E(G)$ increases by at most 1.

Also note that if such a node u was deficient then T actually decreases; if none of these was deficient then the set of deficient nodes did not change. As remarked, this proves the theorem

Example 3.3.3 (China TST 2018)
Two positive integers $p, q \in \mathbf{Z}^+$ are given. There is a blackboard with n positive integers written on it. A operation is to choose two same number a, a written on the blackboard, and replace them with $a+p, a+q$. Determine the smallest n so that such operation can go on infinitely.

3.4 Ignore or redefine

Now, we will alter the problem statement slightly in such a way that the result we need to show does not change, but the process becomes much easier to analyze. In other words, we simplify the process to be analyzed while leaving the aspect of the process that we want to prove something about invariant.

Example 3.4.1
There are n ants on a stick of length one unit, each facing left or right. At time $t = 0$, each ant starts moving with a speed of 1 unit per second in the direction it is facing. If an ant reaches the end of the stick, it falls off and doesn't reappear. When two ants moving in opposite directions collide, they both turn around and continue moving with the same speed (but in the opposite direction).
Show that all ants will fall off the stick in at most 1 second.

The key observation is that the problem does not change if we alter it as follows: when two ants moving in opposite directions meet, they simply pass through each other and continue moving at the same speed. Thus, instead of rebounding, if the ants pass through each other, the only difference from the original problem is that the identities of the ants get exchanged, which is inconsequential.

Now the statement is obvious: each ant is unaffected by the others, and so each ant will fall of the stick of length one unit in at most 1 second.

Example 3.4.2 (USAJMO 2018)
Karl starts with n cards labeled $1, 2, 3, \dots, n$ lined up in a random order on his desk. He calls a pair (a, b) of these cards swapped if $a > b$ and the card labeled a is to the left of the card labeled b. For instance, in the sequence of cards $3, 1, 4, 2$, there are three swapped pairs of cards, $(3, 1)$, $(3, 2)$, and $(4, 2)$.
He picks up the card labeled 1 and inserts it back into the sequence in the opposite position: if the card labeled 1 had i card to its left, then it now has i cards to its right. He then picks up the card labeled 2 and reinserts it in the same manner, and so on until he has picked up and put back each of the cards $1, 2, \dots, n$ exactly once in that order. (For example, the process starting at $3, 1, 4, 2$ would be $3, 1, 4, 2 \to 3, 4, 1, 2 \to 2, 3, 4, 1 \to 2, 4, 3, 1 \to 2, 3, 4, 1$.)
Show that, no matter what lineup of cards Karl started with, his final lineup has the same number of swapped pairs as the starting lineup.

There's almost no way we can control where the cards land and how inversions change after the first turn. Only inversions involving the smallest and largest number are easy to control. The turns are made in successive order, from smallest to largest, and the number that is written plays role. This suggests 1 to modify the number written on the card at each step. Since we are interested in the order of the cards and want to work with largest/smallest, we will just make this card have the largest label, so after each step we will be moving the smallest card and altering it to the largest card, which makes the process more definite. We will alter it by adding n. If we will be able to prove something in this case, then we can repeat it at each step, because all the steps are the same. It turns out that we can, and the main claim is:

Claim – In the modified process, the number of inversions doesn't change at each step.!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Proof. We just have to show that for the first step, because, how I already said, all the steps are the same: We take the smallest number, reflect it and make it the largest. Inversions involving the smallest number are exactly the pairs with the numbers to the left of it. After we reflect it and make it largest, only inversions will be the ones that already existed and the ones with the new largest number with numbers to the right of it, but this quantity is equal to the inversions involving the unmoved smallest number, since we reflected the position. Spamming the claim, we see that in the end of the modified process, we will have the same number of inversions as in the beginning, and the numbers will be the initial numbers incremented by n, so if we decrease them by n, we will not change any of the inversions, so the number of inversions is the same even for the unmodified process.

3.5 Graphs

3.5.1 Diamond lemma

Example 3.5.1
Pirates have an infinite number of chests, numbered in natural numbers. Initially, the first chest contains 2023 *coins. Every day, the captain selects a chest with the number k, in which there are at least 2 more coins than in the next one, and transfers one coin from it to the next. One of these days, the captain won't be able to find such a chest. How could the coins have been distributed at this point?*

3.6 Problems

Problem 3.6.1 (Russia 1995) There are three piles of stones. Sisyphus moves one stone at a time from one pile to another. For each transfer, he receives from Zeus a number of coins equal to the difference between the number of stones in the pile to which he moves the stone and the number of stones in the pile from which he takes it (the stone being moved is not counted in either pile during this calculation). If this difference is negative, Sisyphus must pay Zeus the corresponding amount. (If Sisyphus cannot pay, the generous Zeus allows him to make the move on credit.) At some point, it turned out that all the stones lay in the same piles as they did initially. What is the maximum total profit Sisyphus could have earned by that moment?

Problem 3.6.2 (Russia 2008) In the cells of a 5×5 square, zeros were initially written. Each minute, Vasya chose two cells sharing a common side and either added one to both of their numbers or subtracted one from both. After some time, it turned out that the sums of the numbers in all rows and columns were equal. Prove that this happened after an even number of minutes.

Problem 3.6.3 (IMO 2019) The Bank of Bath issues coins with an H on one side and a T on the other. Harry has n of these coins arranged in a line from left to right. He repeatedly performs the following operation: if there are exactly $k > 0$ coins showing H, then he turns over the k-th coin from the left; otherwise, all coins show T and he stops.

For example, if $n = 3$ the process starting with the configuration THT would be $THT \to HHT \to HTT \to TTT$, which stops after three operations.

Show that, for each initial configuration, Harry stops after a finite number of operations.

Problem 3.6.4 (IMO 2022) The Bank of Oslo issues two types of coin: aluminum (denoted A) and bronze (denoted B). Marianne has n aluminum coins and n bronze coins arranged in a row in some arbitrary initial order. A chain is any subsequence of consecutive coins of the same type. Given a fixed positive integer $k \leq 2n$, Gilberty repeatedly performs the following operation: he identifies the longest chain containing the k^{th} coin from the left and moves all coins in that chain to the left end of the row.

For example, if $n = 4$ and $k = 4$, the process starting from the ordering $AABBBABA$ would be $AABBBABA \to BBBAAABA \to AAABBBBA \to BBBBAAAA \to \ldots$

Find all pairs (n, k) with $1 \leq k \leq 2n$ such that for every initial ordering, at some moment during the process, the leftmost n coins will all be of the same type.

Problem 3.6.5 (APMO 2025) Let $n \geq 3$ be an integer. There are n cells on a circle, and each cell is assigned either 0 or 1. There is a rooster on one of these cells, and it repeats the following operation:

- If the rooster is on a cell assigned 0, it changes the assigned number to 1 and moves to the next cell counterclockwise.
- If the rooster is on a cell assigned 1, it changes the assigned number to 0 and moves to the cell after the next cell counterclockwise.

Prove that the following statement holds after sufficiently many operations:

If the rooster is on a cell C, then the rooster would go around the circle exactly three times, stopping again at C. Moreover, every cell would be assigned the same number as it was assigned right before the rooster went around the circle three times.

Problem 3.6.6 (Serbia 2022) On the board are written n natural numbers, $n \in \mathbb{N}$. In one move it is possible to choose two equal written numbers and increase one by 1 and decrease the other by 1. Prove that in this the game cannot be played more than $\frac{n^3}{6}$ moves.

Problem 3.6.7 (ELMOSL 2013) There are 20 people at a party. Each person holds some number of coins. Every minute, each person who has at least 19 coins simultaneously gives one coin to every other person at the party. (So, it is possible that A gives B a coin and B gives A a coin at the same time.) Suppose that this process continues indefinitely. That is, for any positive integer n, there exists a person who will give away coins during the nth minute. What is the smallest number of coins that could be at the party?

Problem 3.6.8 (ISL 2018) Let a and b be distinct positive integers. The following infinite process takes place on an initially empty board.

1. If there is at least a pair of equal numbers on the board, we choose such a pair and increase one of its components by a and the other by b.
2. If no such pair exists, we write two times the number 0.

Prove that, no matter how we make the choices in (i), operation (ii) will be performed only finitely many times.

Chapter 4

Games

In the olympiads usually we only considered games with ”complete information”. That means that every player (if there are multiple ones) knows fully what he (and other players) can do at any moment of time, so there is no hidden cards or random events. Some examples of such games are chess, tic-tac-toe and checkers.

Chapter 5

Double Counting

Often you find yourself in front of the problem that requires you to find number of ways to do something or to prove that 2 super complicated sums are equal to each other.

In the problems like this we define some number, and try to count it in two different ways. One of the ways is often easily calculatable, which allows us to find the sum very easily.

5.1 Binomial Stuff

One big part of such problems is proving some equalities with binomial coefficients.

Example 5.1.1
Prove that

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}.$$

Suppose you have n white balls, and you want to color k of them black, in how many ways you can do that?

We want to chose k from n, and the order at which we pick the balls does not matter. So the answer (kind of) by definition is $\binom{n}{k}$. So the LHS is done.

Now to the RHS, point at one specific ball (let's say rightmost one), it has 2 possibilities – to be colored or not to be colored.

- If we color it we need to color $k-1$ balls from the leftover $n-1$ balls – $\binom{n-1}{k-1}$.

- If we do not color it, we need to chose all of k balls from the leftover – $\binom{n-1}{k}$.

Adding this two up we get to the initial equality.

Example 5.1.2
Prove that for all $n \in \mathbb{N}$
$$\sum_{k=0}^{n}\binom{n}{k} = 2^n.$$

Again, suppose you have n white balls, and you want to color k of them black. But in this case, k may be varied from 0 0 to n.

Let T be the number of ways to do so.

On one hand, every ball may or may not be colored, so it has 2 options, which are independent of each other, so in total $T = 2^n$.

On the other hand,

- What is the number of ways to color 0 balls – $\binom{n}{0}$.
- What is the number of ways to color 1 balls – $\binom{n}{1}$.
- What is the number of ways to color 2 balls – $\binom{n}{2}$.
- ...

So the total is $T = \dbinom{n}{0} + \dbinom{n}{1} + \cdots + \dbinom{n}{n} = \displaystyle\sum_{k=0}^{n}\binom{n}{k}$.

Example 5.1.3
Prove that
$$\binom{n}{1} + 2\binom{n}{2} + 3\binom{n}{3} + \cdots + n\binom{n}{n} = n \cdot 2^{n-1}.$$

Firstly write LHS as a sum – $\displaystyle\sum_{k=1}^{n} k\binom{n}{k}$. Notice that we can replace k with $\binom{k}{1}$. This may seem to overcomplicate the sum, but actually it provides motivation for what to count.

Suppose you have n white balls, and you want to color one of them black, and some number of them blue.

Let T be the number of ways to do so.

We have two routes to count T:

- Firstly chose the ball which is to be colored black, and the chose some number of balls to be colored blue from the rest.
- Chose which balls we color, and then from this set chose which ball we color black.

In the first way we can chose black ball in n different ways, and the rest is $n-1$ balls from which we chose some, so $T = n \cdot 2^{n-1}$.

In the second,

- If we chose to color only 1 of the balls, there is $\binom{n}{1}$ ways to do so, and we need to chose 1 from the set of 1 balls – $\binom{n}{1}\binom{1}{1}$.
- If we chose to color 2 balls, there is $\binom{n}{2}$ ways to do so, and we need to chose 1 to be colored black – $\binom{n}{2}\binom{2}{1}$.
- ...

So in total we have $T = \sum_{k=1}^{n} \binom{n}{k}\binom{k}{1}$.

You can actually generalize this problem to get $\displaystyle\sum_{k=d}^{n} \binom{n}{k}\binom{k}{d} = 2^{n-d}\binom{n}{d}$.

Example 5.1.4
Prove that

$$\sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n-k}{k} = F_n.$$

In a combinatorical way F_n can be expressed as number of ways to tile $1 \times n$ strip of squares with 1×1 and 1×2 rectangles.

So we just need to prove that LHS also count the same number.

If we use k 1×2 tiles, then we must have $n - 2k$ 1×1 tiles. So we need to arrange these $k + n - 2k = n - k$ tiles in some order. The ways to do so is – $\binom{n-k}{k}$. Summing all of them up we get the identity.

5.2 Clubs

5.3 Problems

Problem 5.3.1 Prove that

$$\sum_{i=0}^{k} \binom{m}{i}\binom{n}{k-i} = \binom{m+n}{k}.$$

Problem 5.3.2 Prove that

$$\sum_{k=m}^{n} \binom{k}{m} = \binom{n+1}{m+1}.$$