We will begin soon.

Some Housekeeping

- Today is only for explaining the solutions for the combinatorics problems in the mock IMO, plus some philosophy.

- I think it would be good if we do two problem solving sessions on Wednesday, both of which focuses on solving IMO problems. But, I want to cancel those sessions if most of you prefer to self-study.

Record the meeting…

Things to Note

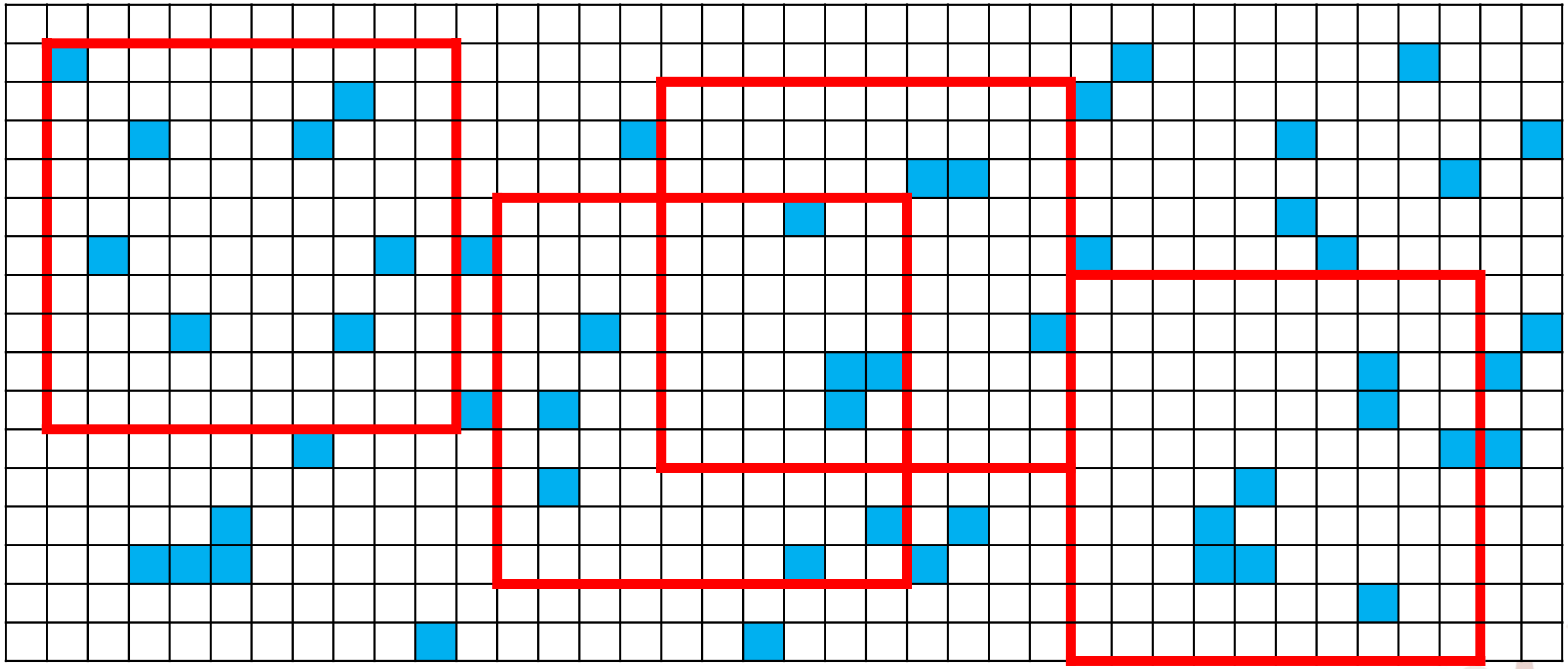
- Understanding the problem statement and what to prove is the most important part. Unwrapping what to do is sometimes not that easy, for example: Phoe Wa Lone problem.

- Fake solving is a common thing in combinatorics. So, always question your results. You should be hard to convince.

- Yes, heuristics are important, but they are not concrete and not rigorous. It is better to spend time on making your heuristics concrete rather than making more heuristics.

Problem 2: One Percent Blue

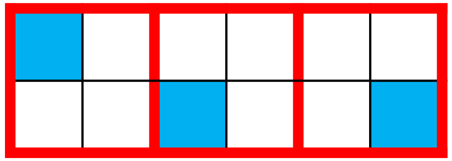
Some cells of an infinite chessboard (infinite in all directions) is coloured blue so that at least one out of 100 cells in any 10×10 rectangular grid is blue. Prove that, for any positive integer n, it is possible to select n rows and n columns so that all of the n^2 cells in their intersections are blue.

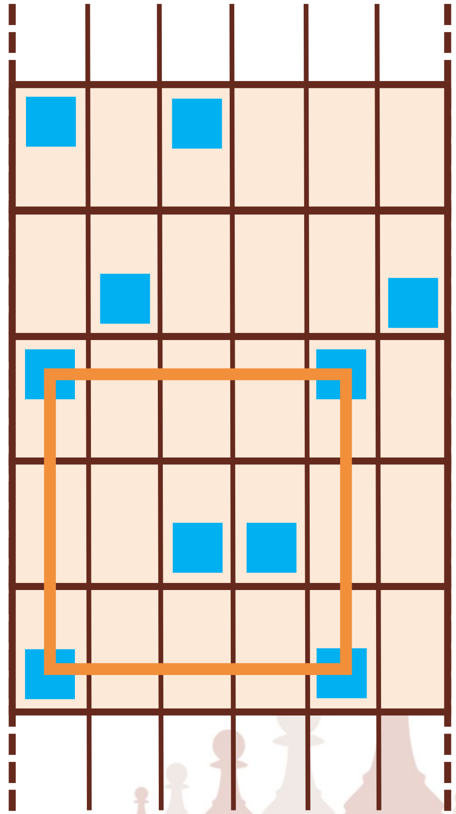


Solution

Let's replace 10×10 with 2×2 first. Let's try to prove for $n = 2$.

- Then, any 2×6 grid contains two blue cells lying in the same row.
- Now, consider a strip of 6 columns.
- This strip can be tiled with infinitely many 2×6 grids each of which contains two blue cells in a row.
- Thus, infinite pigeonhole principle guarantees the desired conclusion for $n = 2$. (In fact, we can even find 2 columns and infinite number of rows.)

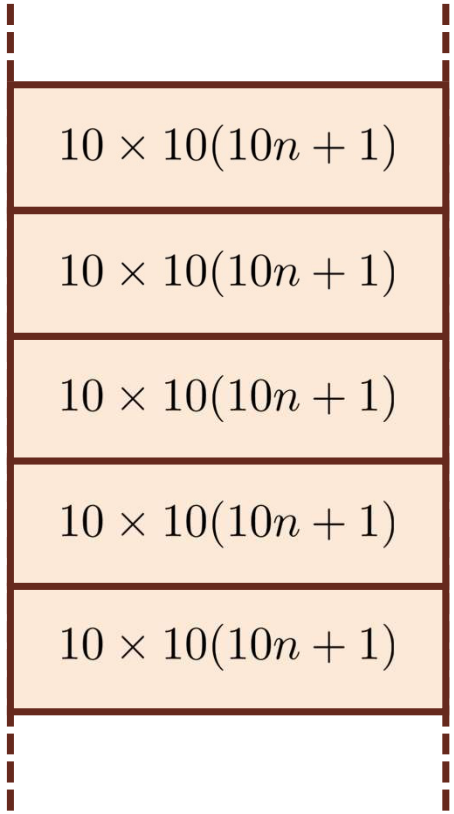




Solution

Now, we can go back to 10×10 and general n.

- Any $10 \times 10(10n+1)$ grid contains a row with n blue cells.
- Now, consider a strip of $10(10n+1)$ columns. This strip can be tiled with an infinite number of $10 \times 10(10n+1)$ grids.
- By infinite pigeonhole principle, we can now choose n rows (in fact, an infinite number) with same "column pattern".



Note: Finite version of pigeonhole principle is necessary to have a finite number of "column patterns".

Problem 7: Phoe Wa Lone's Game

Let $n \geq 2$ be a positive integer. A total of $2n$ balls are coloured with n colours so that there are two balls of each colour. These balls are put inside n cylindrical boxes with two balls in each box, one on top of the other. Phoe Wa Lone has an empty cylindrical box and his goal is to sort the balls so that balls of the same colour are grouped together in each box. In a *move*, Phoe Wa Lone can do one of the followings:

- Select a box with exactly two balls and reverse the order of the top and the bottom balls.
- Take a ball b at the top of a non-empty box and either put it inside an empty box or put it in the box only containing the ball of the same colour as b.

Find the smallest positive integer N such that for any initial placement of the balls, Phoe Wa Lone can always achieve his goal using at most N moves in total.

What do we have to prove?

Find the smallest positive integer N such that for any initial placement of the balls, Phoe Wa Lone can always achieve his goal using at most N moves in total.

Let's say we want to show that $N = 1000$ for example. Then, we have to show the following:

- Phoe Wa Lone can always achieve his goal using at most 1000 moves in total.

This gives $N \leq 1000$.

- Phoe Wa Lone cannot always to achieve his goal using at most 999 moves in total.

This gives $N \geq 1000$.

- The second condition is equivalent to "there is an initial configuration in which Phoe Wa Lone needs at least 1000 moves."

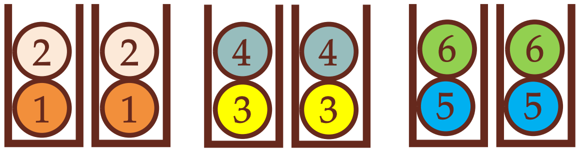
Getting the logic and sketch right is probably the most important thing!

The Solution: Part I

The answer is $N = n + \left\lfloor \frac{n}{2} \right\rfloor$.

The initial configuration that needs $n + \left\lfloor \frac{n}{2} \right\rfloor$ moves

- For n even, consider the following:

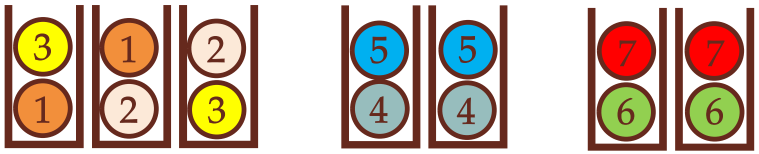


$\cdots$

We need 3 moves to sort each pair:

- One move to sort colour 1
- One move to sort colour 2
- One move into empty box

- For n odd, consider the following:



$\cdots$

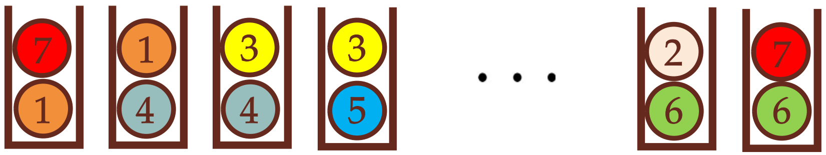
Similar reason as above.

- 4 moves to sort first 3 colours,
- 3 moves to the remaining pairs.

The Solution: Part II Step 1

Now, we prove that any initial configuration can be solved within $n + \left\lfloor \frac{n}{2} \right\rfloor$ moves. Suppose WLOG that none of the boxes are already sorted.

Call a sequence of boxes $B_1, B_2, B_3, \ldots, B_k$ a *cycle* if B_i and B_{i+1} share a same coloured ball for each $i = 1, 2, 3, \ldots, k-1$, and B_1 and B_k share a same coloured ball.



Step 1: Reducing the Problem

- The boxes can be partitioned into disjoint cycles. ← Just RUST
- If we can sort a cycle of k boxes using $k + \left\lfloor \frac{k}{2} \right\rfloor$ moves, then we are done!

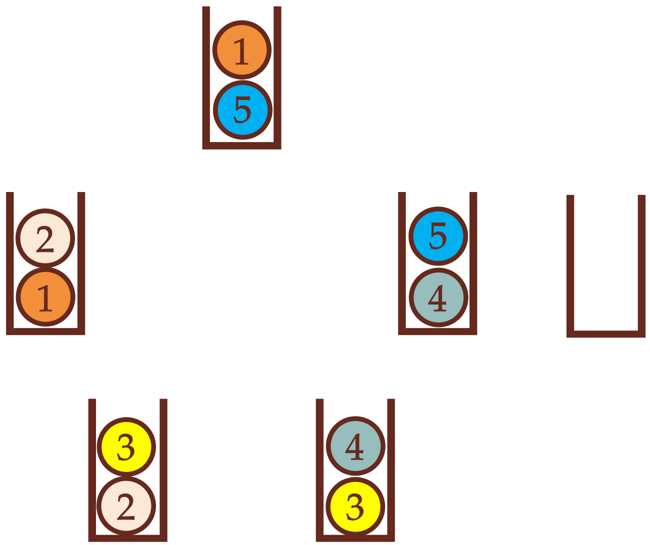
The Solution: Part II Step 2

Step 2: Finding Easy Configurations

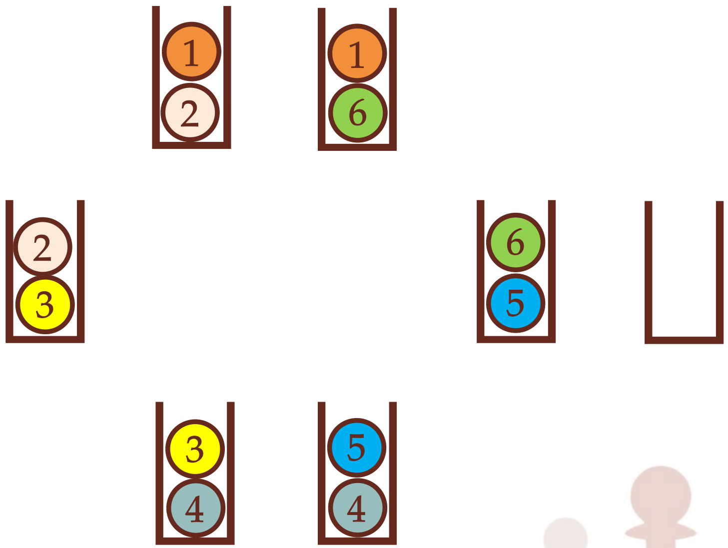
Note that this is the best we can hope.

If our cycle has the following forms, we can sort them in just $k + 1$ moves.

- All the top balls have distinct colours



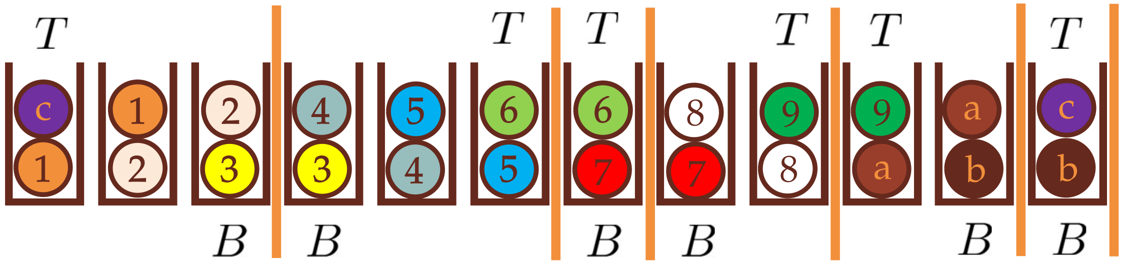
- Only one pair of top balls have same colour.



The Solution: Part II Step 3

Step 3: Reduction to Easy Configuration

Note that top pairs and bottom pairs alternate along the cycle.



Critical Observation: Reversing alternating chunks makes all the top balls distinct.

Thus, we can make all top balls distinct using $\left\lfloor \frac{k}{2} \right\rfloor$ moves.

Thus, we can make all top balls except one pair distinct using $\left\lfloor \frac{k}{2} \right\rfloor - 1$ moves. Bingo!

The Solution: Summary

- Step 1: Phoe Wa Lone partitions the boxes into disjoint cycles.
- Step 3: For each cycle with k boxes, use at most $\left\lfloor \frac{k}{2} \right\rfloor - 1$ to make all but one pair of top balls distinct.
- Step 2: Sort the resulting cycle with $k+1$ moves.

Things to Note (again)

- Understanding the problem statement and what to prove is the most important part. Unwrapping what to do is sometimes not that easy, for example: Phoe Wa Lone problem.

- Fake solving is a common thing in combinatorics. So, always question your results. You should be hard to convince.

- Yes, heuristics are important, but they are not concrete and not rigorous. It is better to spend time on making your heuristics concrete rather than making more heuristics.

Some Philosophy

- Make sure to fully understand the problem statement and the logic flow of what you need to prove. You don't want to spend 4 hours thinking about something that isn't even asked for.
- Two-part problems are quite popular these days. The construction part is usually easier. Try out as many constructions as you can. Don't make conjectures too early.
- Study the small cases, or simpler variants of the problem. They help you understand/reduce /or even solve the main problem.
- Try to spot small observations. Things like "… is always …" or like "if …, then …" . Usually, they help you reduce the problem into a simpler one.
- Heuristics are good, but don't rely too much on them. Concrete results are much better and reliable than heuristics.

Questions to Ask Yourself

- Induction…
- Is the problem made up with many smaller pieces?
- Have you tried rephrasing the problem? (into graph language for example)
- Any observations that you have not tried to expand on?
- (for grid problems) How many colourings have you tried? Have you tried sub-dividing into multiple grids?
- (for process problems) Have you tried assigning weights?

That’s it for this lecture.

See you soon!