EGOI 2025 Editorial - Monster Go

Task authors: Dario Martinez Ramirez, Manuel Torres Cid

July 16, 2025

In this problem, you have to output N lists each consisting of 12 distinct numbers from 0 to 49. A list wins if all its numbers have been enumerated first. You have to make sure that no matter in which order the numbers are enumerated, there will always be a unique winner. A general observation here is that when you have two lists A and B, for each number k in $A \cap B$, you also need a list that is a subset of $A \cup B \setminus \{k\}$ - this makes sure that we can never run into a situation where A and B both only still need number k for winning.

What is special about this problem is that the test cases are public - there are many different approaches on how to solve some test cases, it is even possible to solve the smaller ones by hand to score some points. In the following, we explain some approaches on how to construct such lists and then show how to combine those approaches into a full solution. By only finding some but not all of these constructions, one can obtain a partial solution.

Disjoint Lists. The simplest way to construct such a set of lists is to just use disjoint lists. This approach can solve $N = 1, N = 2, N = 3, N = 4$.

Rotations. We can create 13 lists using 13 numbers: create list i by removing the i-th number from the set. There will be a unique winner now: assume that the number k is the last number that makes 2 lists A and B win at the same time: then there is also a list $(A \cup B) \setminus \{k\}$ which would then win before A and B, so we have a contradiction. This idea can solve $N = 13$, and by iteration we can also get $N = 26$ and $N = 39$

Generalized Rotations. We can generalize the idea from before: Take any divisor d of 12. Define

$$S_1 = \{1, \dots d\}, S_2 = \{d + 1, \dots 2d\}, \dots S_{n/d+1} = \{13, \dots 12 + d\}$$

Then, construct list

$$l_j = (\bigcup S_i) \setminus S_i$$

. The correctness proof is the same as for the simple rotation. When choosing $d = 1$, we obtain back the simple rotation. This idea can solve cases $N = 12/d + 1$ for any divisor d of 12 (and can again be iterated)

Generalizing further. Note that we can also add additional subsets to a generalized rotation: For some positive integer k and a divisor d, define

$$S_1 = \{1, \dots, d\}, \dots, S_{n/d+1} = \{13, \dots 12 + d\},$$
$$\dots, S_{n/d+k} = \{13 + (k - 1)d, \dots 12 + k \cdot d\}$$

Then, we can create $\binom{n/d+k}{k}$ lists, each one containing all numbers in all of the subsets except k of them.

Combining approaches. More solutions can be obtained by combining the approaches from before: Define a *block* of size N with cost c as a solution for N that uses c distinct numbers.

If we now want to construct a solution for a new N, we have to find blocks $(N_1, c_1) \dots (N_k, c_k)$ such that $N_1 + \dots + N_k = N$ and $c_1 + \dots + c_k \leq 50$. There are two ways to find such a combination:

Note that since every block has a cost of at least 12, we can combine at most 4 blocks. Thus, we can iterate through all possible combinations of four blocks to find a solution. Instead, it is also possible to use a knapsack DP.