

Preliminaries

We are interested in impartial games under the normal play convention which must end. These are games where the two players have the same move set from a given position and where the winning player is the one who plays the final move. We can define each game as a set of games $G = \{G_1, G_2, \dots\}$. This means that G is the game where the current player chooses one of the G_i -s and “moves” to it, then the second player plays in the chosen G_i . For brevity, we will write this as $G = \{G_i\}$, where the i implicitly indexes through all the possible games $G_1, G_2, \dots$. Notice that this set need not be finite. Furthermore, we care about combining games. Given games G and G' , we say that $G + G'$ is the game where the current player can either play in G or in G' (but not both or neither). Notice that a game of Nim is such a combination of several Nim heaps. Formally, $G + G' = \{G_i + G'_j\} \cup \{G + G'_i\}$. Additionally, we say that a game G is winning, if under optimal play the first player wins, and we call it losing, otherwise. Finally, we say that two games G and G' are equivalent, which we write as $G \approx G'$, if and only if for all other games H , $G + H$ is winning if and only if $G' + H$ is winning.

Now, we familiarize ourselves with the Sprague-Grundy theorem. It states that every game G is equivalent to a single Nim heap $*n$ with n pebbles. While the proof is somewhat involved, we give the recursive procedure for computing this n . First, the empty game $\emptyset \approx *0$. Then, for all other games G , we have that $G = \{G_i\} \approx \{ *n_i \} \approx * \text{mex}\{n_i\}$; $\text{mex } A$ is the minimum excluded value of A , the smallest number m , such that $m \notin A$. We can further show that $*n + *m \approx *(n \oplus m)$, where $n \oplus m$ is the XOR of n and m .

Subtask 1: 8 points, 8 total

The first subtask is just the standard game of Nim. The game is winning, iff the XOR of the heap sizes is non-zero. This follows from the above. It can also be directly proven by showing that from any non-zero position, you can reach a zero position, and from any zero position, you can only reach non-zero positions.

Subtask 2: 11 points, 19 total

We compute the XOR of all pebble heap sizes and the parity of the number of stones. The game is winning, iff the XOR of the pebble sizes is non-zero or the parity is odd. This follows from the next part.

Subtask 3: 12 points, 31 total

We can analyze these games either by directly looking at winning and losing positions or by extending the Sprague-Grundy theorem to them. First, we will follow the direct approach. We look at the XORs of just the pebble heaps S' and the XOR of the stone heaps B' . Note that we treat a mixed heap in the same way as two heaps: one with only pebbles and one with only stones. We claim that a position is losing, iff $B' = 0$ and $S' = 0$. First, consider every winning position. If $B' = 0$ and $S' \neq 0$, then by using the standard Nim approach we can reach a losing position. Similarly, if $B' \neq 0$ and $S' = 0$ (since $K = 0$ we can remove the stones without adding any pebbles). Then, if $B' \neq 0$ and $S \neq 0$, we choose our stone move as with the previous case, but we also take all pebbles from the chosen heap; after that we add S'' pebbles, where S'' is the XOR of all other pebble heaps. Therefore, from all winning positions, we can reach a losing position. Finally, from a losing position ($B' = S' = 0$), we have two choices: if we take only pebbles, then the pebble XOR we leave will be non-0, and, if we take some stones, then the stone XOR we leave will be non-0; in either case, we cannot reach a losing position. Therefore, our claim is correct.

The alternative approach is to naturally extend the Sprague-Grundy theorem to transfinite ordinals of the form $\omega B + S$ and to show $*(\omega B + S) + *(\omega B' + S') \approx *(\omega(B \oplus B') + (S \oplus S'))$. Here $\omega B + S$ is the game that corresponds to B stones and S pebbles with $K = 0$.

Subtask 4: 18 points, 49 total

This subtask requires the understanding needed for the next one without coding up anything complicated. It allows us to find the equivalent transfinite Nim heaps for the first 15 interesting heaps by hand and hard-code them. Thus, we leave the mathematics to the next part.

Subtask 5: 18 points, 67 total

We adopt the notation $(B, S)_K$ to mean a heap with B stones and S pebbles under a given K . Our goal is to find X and Y such that $(B, S)_K \approx (X, Y)_0 = *(\omega X + Y) = *f_K(B, S)$. We do this recursively using:

$$\begin{aligned} (B, S)_K &= \{(B', S')_K \mid B' < B \wedge S' \geq BK\} \cup \{(B, S')_K \mid S' < S\} \\ &\approx \{ *f_K(B', S') \mid B' < B \wedge S' \geq (B - B')K \} \cup \{ *f_k(B, S') \mid S' < S \} \\ &\approx * \text{mex}(\{f_K(B', S') \mid B' < B \wedge S' \geq (B - B')K\} \cup \{f_k(B, S') \mid S' < S\}) \end{aligned}$$

Therefore, we just need to recursively (with memoization) compute the interesting values of f by computing such MEX-es, using the fact that ordinals of the form $\omega X + Y$ are ordered lexicographically. Additionally, to prevent having to iterate through an infinite number of values, we need to notice (or prove by induction) that when $S \geq BK$, $(B, S)_K \approx \omega B + (S - BK)$. Therefore, there are $O(B^2 K)$ interesting heaps and for each there are $O(B^2 K)$ possible interesting moves, which we need to sort and compute the MEX of. The total complexity of this naïve approach is $O(B^4 K^2 \log BK)$.

To illustrate, we give an example with $K = 2$ for computing the value of the first empty cell in the table below (rows are indexed by B and columns – by S):

$\omega 0 + 0$	$\omega 0 + 1$	$\omega 0 + 2$	$\omega 0 + 3$	$\omega 0 + 4$	$\omega 0 + 5$
$\omega 0 + 0$	$\omega 0 + 1$	$\omega 1 + 0$	$\omega 1 + 0$	$\omega 1 + 2$	$\omega 1 + 3$
$\omega 0 + 0$	$\omega 0 + 1$	$\omega 0 + 2$	$\omega 0 + 3$	$\omega 2 + 0$	$\omega 2 + 1$
$\omega 0 + 0$	$\omega 0 + 1$	$\omega 0 + 4$			

First, we want to compute $f_2(3, 3)$. We list the reachable heaps from $(3, 3)_2$ (where ... means the list of all larger heaps with the same B): $(0, 6)_2, \dots, (1, 4)_2, \dots, (2, 2)_2, \dots, (3, 0)_2, (3, 1)_2, (3, 2)_2$. Now we give their values: $\omega 0 + 6, \dots, \omega 1 + 2, \dots, \omega 0 + 2, \omega 0 + 3, \omega 2 + 0, \dots, \omega 0 + 0, \omega 0 + 1, \omega 0 + 4$. The smallest value not in this list is $\omega 5 + 0$, so that is the value of the first unfilled cell.

Subtask 6: 10 points, 77 total

Now we need to notice that the interesting moves from $(B, S)_K$ are the same ones as from $(B, 0)_K$ plus the moves to $(B, 0)_K, (B, 1)_K, \dots, (B, S - 1)_K$. Therefore, we compute $f_K(B, S)$ for all $S < BK$ by sorting the interesting values of the (multi)set $\{f_K(B', S') \mid B' < B \wedge S' \geq (B - B')K\}$, iterating through it and setting the “gaps” to be values of $f_K(B, 0), f_K(B, 1), \dots, f_K(B, BK - 1)$. Again, we need to use the $(B, S)_K \approx \omega B + (S - BK)$ fact. Therefore, for each B we only need to sort and iterate through $O(B^2 K)$ interesting values, and thus the total complexity is $O(B^3 K \log BK)$.

Again, we illustrate by computing the values in the entire $B = 3$ row of the table above. The values reachable from $(3, 0)_2$ are: $(0, 6)_2, \dots, (1, 4)_2, \dots, (2, 2)_2, \dots$. Then, their values, in ascending order, are: $\omega 0 + 2, \omega 0 + 3, \omega 0 + 6, \dots, \omega 1 + 2, \dots, \omega 2 + 0, \dots$. Now we need to find the “gaps” below $\omega 3 + 0$; they are: $\omega 0 + 0, \omega 0 + 1, \omega 0 + 4, \omega 0 + 5, \omega 1 + 0, \omega 1 + 1$. Therefore, these are the first six values for $B = 3$.

Subtask 7: 11 points, 88 total

Let us say that $R_K(B, S)$ is the (multi)set of the interesting f_K values of moves from $(B, S)_K$. First, we need to prove (by induction) that it is a set and not just a multiset, meaning that all its elements are unique. This follows from the fact that if $(B', S')_K$ and $(B'', S'')_K$ are interesting moves of $(B, S)_K$ either $(B', S')_K$ is an interesting move of $(B'', S'')_K$ or vice versa. WLOG assume the former. Then, from $f_K(B', S') = \text{mex } R_K(B', S')$ and $f_K(B'', S'') \in R_K(B', S')$, it follows that $f_K(B', S') \neq f_K(B'', S'')$. Thus, $R_K(B, S)$ contains no duplicates.

Now we examine $R_K(B - 1) \setminus R_K(B)$. The only heaps unreachable from $(B, 0)_K$ but reachable from $(B - 1, 0)_K$, are of the form $(B', S')_K$ where $B' < B$ and $(B - B' - 1)K \leq S' < (B - B')K$. Notice that are BK such heaps. By the property above, their values are all distinct from the values of all other heaps reachable from $(B - 1, 0)_K$. Therefore, all their values are “gaps” in $R_K(B)$. However, by the definition of $f_K(B - 1, S)$, it follows that $R_K(B - 1) \cup \{f_K(B - 1, S') \mid S' \geq 0\}$ has no “gaps”, meaning that it is equal to the set $\{\omega X + Y \mid 0 \leq X \leq B - 1 \wedge Y \geq 0\}$ (otherwise all “gaps” would be included as values of $f(B - 1, S')$ for some S'). Thus, we conclude that the only “gaps” of $R_K(B)$ are the BK values in the given form, which finally proves that $(B, S)_K \approx \omega B + (S - BK)$ for $S \geq BK$. Using this, we can directly set the values of the heaps for each B as the sorted values of the heaps in the given form, $B' < B$ and $(B - B' - 1)K \leq S' < (B - B')K$. The total complexity is $O(B^2 K \log BK)$.

Again, we illustrate this by computing the values of the $B = 3$ row in the table. The values on the relevant “diagonal” are: $\omega 0 + 4, \omega 0 + 5, \omega 1 + 0, \omega 1 + 1, \omega 0 + 0, \omega 0 + 1$. Thus, these (but sorted) are the first six values of the $B = 3$ row.

Subtask 8: 12 points, 100 total

Finally, we need to see that the only “truly interesting” values of f are ones where K divides S . This is because, if K does not divide S , then $f_K(B, S) = f_K(B, S - 1) + 1$. Intuitively, this means that the values of f come in chunks of K sequential ones. This can easily be proven by induction on B , since the values for $S \geq BK$ are all sequential and the “diagonals” from which we are taking the interesting values of f (where $(B - B' - 1)K \leq S' < (B - B')K$) is K -aligned. Thus, we need to every K -th value of f , by looking only one interesting point per B' (for a given B). Therefore, the “truly interesting” values of f for a given B are the sorted set of “truly interesting” values of heaps of the form $(B', (B - B' - 1)K)_K$. There are B such values for a given B and thus the total complexity is $O(B^2 \log B)$.

Note that the structure of the table of “truly interesting” values is invariant of K . Its first few values (the ones needed for subtask 4) are given below (rows are indexed by B and columns – by S/K):

$\omega 0 + 0K$	$\omega 0 + 1K$	$\omega 0 + 2K$	$\omega 0 + 3K$	$\omega 0 + 4K$	$\omega 0 + 5K$
$\omega 0 + 0K$	$\omega 1 + 0K$	$\omega 1 + 1K$	$\omega 1 + 2K$	$\omega 1 + 3K$	$\omega 1 + 4K$
$\omega 0 + 0K$	$\omega 0 + 1K$	$\omega 2 + 0K$	$\omega 2 + 1K$	$\omega 2 + 2K$	$\omega 2 + 3K$
$\omega 0 + 0K$	$\omega 0 + 2K$	$\omega 1 + 0K$	$\omega 3 + 0K$	$\omega 3 + 1K$	$\omega 3 + 2K$
$\omega 0 + 0K$	$\omega 0 + 1K$	$\omega 0 + 3K$	$\omega 1 + 1K$	$\omega 4 + 0K$	$\omega 4 + 1K$
$\omega 0 + 0K$	$\omega 0 + 2K$	$\omega 0 + 4K$	$\omega 1 + 2K$	$\omega 2 + 0K$	$\omega 5 + 0K$