SRS document

Zhraa Zaid 206604027

Ayman Younis 207054354

In our project which is a math gaming project we will discuss **The wonders of Grundy's game,** , we will focus on **A Game of Nim** .

*providing and explaining the mathematical concepts behind Grundy's game which will mainly revolve around the Sprague-Grundy theorem .

*explaining the XOR addition strategy.

*providing an algorithm and Implementation for the classical game of Nim.

*explain applications and variations for this game.

First lets clarify some essential concepts .

Impartial Games

An impartial game is a two-player game where the players alternate making moves. The moves available to a player at any given time must be independent of whose turn it is. Players will alternate taking moves until they reach a terminal position, where one of the players is declared the winner.

Grundy's game

this game is an impartial combinatorial game played by two players. The starting configuration is a single heap of objects, and the two players take turn splitting a single heap into two heaps of different sizes. The game ends when only heaps of size two and smaller remain, none of which can be split unequally. The game is usually played as a normal play game, which means that the last person who can make an allowed move wins.

Sprague-Grundy Theorem

Impartial games can be analyzed using the Sprague-Grundy theorem. They key idea here is to define the Grundy number of a position p in the game. The Grundy number of a position p is defined recursively as follows. It is the smallest non-negative integer, r, such that no successor position has a Grundy number equal to r. By this definition if the position is terminal, then the Grundy number is 0. The successor positions of a position is the set of all positions that can be reached by making a single move from the current position. As an example, if we are currently in a position where we can make only one move, and the position we can move to is terminal, then the Grundy number of the current position is 1 (it is not 0, because we have a successor position whose Grundy number is 0). Here is pseudocode for computing the Grundy number of a position.

The Sprague-Grundy theorem allows us to solve so-called composite games. For impartial games, where it is the case that a player loses if he has no move available, the theorem tells us that a player loses if the xor of the Grundy numbers of the position in each subgame is equal to 0.

Game of Nim:

Nim is a strategy mathematical game in which two players take turns removing items from different heaps. A player must remove at least one object on each turn and can remove any number of objects, provided they all come from the same heap. The goal of the game is to be the player with the last item removed. Typically, Nim is played as a misère game, in which the player loses to take the last object. Nim can also be played as an ordinary game in which the player who takes the last object wins. Since the last move is a winning move in most games, this is called normal play; Nim is usually played such that the last move loses.

Sample Gameplay :

Consider two players, Alice and Bob playing the game of Nim:

Two players: Alice and Bob

Sizes of heaps Moves

A B C

3 4 5 Bob takes 2 from A

1 4 5 Alice takes 3 from C

1 4 2 Bob takes 1 from B

1 3 2 Alice takes 1 from B

1 2 2 Bob takes entire A heap, leaving two 2s.

0 2 2 Alice takes 1 from B

0 1 2 Bob takes 1 from C leaving two 1s. (In misère play he would take 2 from C leaving (0, 1, 0).)

0 1 1 Alice takes 1 from B

0 0 1 Bob takes entire C heap and wins.

0 0 0 Game ends. Bob wins

Algorithm :

The binary digital sum of the heap sizes, that is, the sum (in binary) neglecting all carries from one digit to another is the key to the theory of the game. Often known as as "exclusive or" is this process (xor).

Truth table of the XOR or Nim-sum operation:

Output is 1 if there are odd number of ones in the input

0 if there are even number of ones in the input

Consider it for 2 inputs:

Input_1	Input_2	Output
0	0	0
0	1	1
1	0	1
1	1	0

The winning strategy in normal play is to finish every move with a 0.0 nim-sum. If the nim-sum is not zero before the move, this is always achievable. If the nim-sum is zero, then if the other player does not make a mistake, the next player loses.

So we need to know which step to make; To find that, let X be the nim-sum of all the heap sizes. Find a heap where the nim-sum of X and heap-size is less than the heap-size - the winning strategy is to play in such a heap, reducing that heap to the nim-sum of its original size with X.

As a particular simple case, if there are only two heaps left, the strategy is to reduce the number of objects in the larger heap to make the heaps equal. After that you can make the same move on the other heap, no matter what move your opponent makes, ensuring you take the last object.

And We will explain that more in advanced stages with a code implementation and different examples as we said at the beginning.