Divide to One (divide21 or /21)

A Strategic Game of Digit Manipulation and Divisibility

Jacinto Jeje Matamba Quimua

September 2025

Abstract

Divide21 is a mathematical game designed to foster logical and strategic thinking, improved memory and pattern recognition through digit manipulation and division. Players alternate turns modifying digits and attempting divisions to reduce the number to 1 or reach a score greater than or equal to $9 \times$ (the number of digits in the initial number). This paper introduces the game mechanics, explores its strategic complexity, and raises open questions about optimal play and computational analysis.

Introduction

This paper introduces a new mathematical game called divide21 (or Divide to One), which blends number theory, game theory, and algorithm decision making.
The game also cultivates mental math, logical reasoning and strategic thinking, making it both educational and intellectually stimulating. It is intended to present an engaging challenge that fosters curiosity, deeper exploration and research.
By embedding mathematical logic into a game format, it encourages intuitive exploration, pattern recognition, and decision making skills that are essential not only in mathematics but also in computer science, cryptography, and algorithm design.
Moreover, the structure of the game introduces players to the tension between short-term gains and long-term strategy, a concept central to the game theory and optimization. This makes divide21 not just a learning tool, but a platform for deeper mathematical inquiry and AI experimentation.
This paper presents the rules and mechanics of divide21, gives illustrative examples of gameplay, showing strategic choices and outcomes, it also provides an analysis of the complexity of the game, some open questions related to the optimal play, game termination, and computational difficulty are also presented.

Game Rules

Initial Setup

A random positive integer **X** is generated, such that:

- It has more than one digit, which is determined before the game starts.
- All of its divisors (except 1) must also have more than one digit.

Player Actions

Players alternate turns, and on each turn may:

- Change a single digit of **X**, such that:
    - **X** does not become 0 or 1
    - All ten digits (0-9) must be used at a given index before any of them is reused.
    - Leading zeros are not allowed; if they appear, they cannot be changed.
- Attempt division of **X** by a one-digit number (2-9):
    - If successful, the divisor is added to the player's score.
    - If unsuccessful, the divisor is subtracted from the score.
    - If a player misses a valid division opportunity, the largest one-digit factor of **X** is subtracted from the player's score.

After Division

- The quotient becomes the new **X**, and the player may keep dividing it to get extra points.
- Digits in the quotient cannot be reused at their respective index, unless all ten one-digit numbers (2-9) have been used.
- The timer does not reset.

Win and Loss Conditions

- Win: Reach a quotient 1 or accumulate $9 \times$ (number of digits of **X**) points or more.
- Loss: Run out of time or drop to $-9 \times$ (number of digits of **X**) points or fewer.

Examples

Here are a few examples of the game, with the design from the web application I have created, which are current at the time of this writing.

Example 1:

Player: **41** → change digit 1 to 2 → **42**

Computer: 42 ÷ 7 = 6 (+7 points) → **6**

Computer: 6 ÷ 6 = 1 (+6 points) → **1** (Computer wins by quotient 1)

Example 2:

Player: **493** → change digit 9 to 0 → **403**

Computer: 403 → change digit 3 to 9 → **409**

Player: 409 → change digit 4 to 3 → **309**

Computer: 309 ÷ 3 = 103 (+3 points) → **103**

Computer: 103 → change digit 0 to 7 → **173**

Player: 173 → change digit 3 to 2 → **172**

Computer: 172 ÷ 4 = 43 (+4 points) → **43**

Computer: 43 → change digit 4 to 1 → **13**

Player: 13 → change digit 3 to 7 → **17**

Computer: 17 → change digit 7 to 4 → **14**

Player: 14 ÷ 2 = 7 (+2 points) → **7**

Player: 7 ÷ 7 = 1 (+7 points) → **1** (Player wins by quotient 1)

Strategy and Complexity

In game theory and artificial intelligence, the branching factor represents the average number of legal moves available from a given game state. It quantifies how many possible actions a player can take at each decision point. Formally, if each state s_i in a game has b_i legal moves, and there are S total states, the average branching factor is defined as $\bar{b} = \frac{1}{S}\sum_{i=1}^{S} b_i$. A higher branching factor indicates that the game tree expands more rapidly, leading to exponential growth in the number of possible outcomes as the game progresses. This makes finding optimal strategies computationally more challenging, as the search space increases exponentially with depth ($O(b^d)$, where d is the average game length). For comparison, Tic-Tac-Toe has a branching factor of about 4, Chess about 35, and Go (19×19) roughly 250–300. In Divide21, the branching factor grows approximately linearly with the number of digits n in the current number, estimated by $B(n) \approx 9n + 8$, which is based on up to 9 alternate digit choices per position and up to 8 possible one-digit divisors. Therefore its branching factor surpasses that of Chess when $n \geq 4$ and even exceeding that of Go for $n \geq 33$.

Divide21 has a rich strategic space and several levels of complexity because of its high branching factor. Each state of the game is defined by:

- the value of **X**
- tracking list of available (allowed) digits at each index
- the players' scores
- the turn order (the player with the current turn)

At each turn, the player must choose which digit to modify and what new digit to use under the digit-history constraints. Furthermore, once **X** becomes divisible by a one-digit number (2-9), the player must then decide whether to divide it immediately and earn points (and maybe even start a chain of divisions that lead to 1) or miss that division intentionally and instead make a digit change, which causes point deduction, but gives a greater advantage in the future. This is a key tension in divide21: greedy scoring vs long-term positioning. Dividing by the largest one-digit factor may result in the most points in the short term, however, it might lead to a new number with fewer divisor options or unlock easy paths for the opponent.

Open questions

1. Is dividing **X** (the initial number) by its largest possible one-digit divisor (factor) in order to maximize points per division always the optimal strategy to win the game?
2. How does the initial value of **X** influence the outcome of the game? How does it (or does not) give an advantage to the player with the first turn?
3. Does the initial value of **X** ever forces the player with the first turn to make a digit change that will make **X:**

    a. A number with a one digit factor, thus giving the following player the opportunity to get points?
    b. A number with a one digit factor that immediately leads to a series of divisions until the quotient is 1 or the player has 9 $\times$ (number of digits of **X**) points or more?
4. If all players always play optimally:
    a. Will the games always be finite? Can loops occur?
    b. Is it possible to always win the game only by one method? (getting the quotient to 1 or getting 9 $\times$ (number of digits of **X**) points or more)?
    c. What's the maximum number of primes that **X** can be during the game? Can we list them in advance before the first move is made?
5. For a given number of digits in **X:**
    a. How many possible games are there?
    b. What are the longest and shortest possible games?
    c. What is the computational complexity of finding an optimal path?
    d. What is the maximum number of players in the game so that it is guaranteed that everyone gets at least one turn, assuming optimal play?

Conclusion

Divide21 is a unique mathematical gameplay, combining divisibility puzzles, digit manipulation, and strategic decision making. It offers a platform for both gameplay and deeper mathematical exploration, with potential applications in education, AI research (Symbolic AI benchmark), and discrete mathematics.

References

Shannon, C. E. (1950). *Programming a Computer for Playing Chess*. *Philosophical Magazine*, 41(314), 256–275.

van den Herik, H. J., Uiterwijk, J. W. H. M., & van Rijswijck, J. (2002). *Games Solved: Now and in the Future*. *Artificial Intelligence*, 134(1–2), 277–311.

Silver, D., Huang, A., Maddison, C. J., et al. (2016). *Mastering the Game of Go with Deep Neural Networks and Tree Search*. *Nature*, 529(7587), 484–489.

Knuth, D. E., & Moore, R. W. (1975). *An Analysis of Alpha–Beta Pruning*. *Artificial Intelligence*, 6(4), 293–326. *(for the computational complexity discussion)*

Play Online

I have developed a working version of divide21, available at: www.divide21.com

Copyright