

Mainstream AI Agents in a Logic Number Puzzle Contest

several agents are already able to solve surprisingly complex puzzles

Patrick Min

version 1.0

February 6, 2026

Abstract

Many popular AI Agents use “Large Language Models” (i.e. statistical models of language), so perform typically poorly on tasks involving calculation and logic reasoning. Solving a “Calcudoku” logic/number puzzle is an example of such a challenging task. We tested 10 popular free mainstream AI Agents by asking them to solve Calcudoku puzzles in increasing order of difficulty. The tested agents were (in alphabetical order): ChatGPT, Claude, Copilot, Deepseek, Gemini, Grok, LeChat, Meta AI, Perplexity, and Qwen. The more advanced modes that can create and execute code “on the fly” clearly outperform the simpler (“Fast”) options. However, of the advanced modes only three were actually impressive: **overall winner was Google Gemini “Thinking”, with xAI Grok “Expert” second, and Microsoft Copilot third.**

Keywords: AI, AI Agents, LLM, number puzzles, logic puzzles, calcudoku

1 Introduction

Since the launch of ChatGPT the popularity of AI agents based on “Large Language Models” has exploded. Because they are in essence probabilistic text generators, they tend to struggle with tasks requiring a single exact answer, for example math problems and logic puzzles. Recently, researchers have tried to address this by adding specialized “reasoning” models and adding automatic programming capabilities.

Calcudoku puzzles are a good benchmark for testing this type of performance: solving them requires a combination of math and logic. See the next section for a description of the Calcudoku puzzle.

We tested 10 “mainstream” freely available AI agents by asking them to solve Calcudoku puzzles. The puzzles were presented in increasing order of difficulty. In each “round”, if an agent failed to solve the puzzle it was eliminated, until there was one left standing.

The overall winner was Google Gemini “Thinking”. xAI Grok “Expert” came second, and Microsoft Copilot “Smart” and “Thinking” shared third. Qwen3-Max “Thinking” was fourth, and DeepSeek “DeepThink” came fifth, but was clearly the slowest. The worst performers were Google Gemini “Fast”, xAI Grok “Fast”, Meta AI “Fast”, Mistral LeChat “Deep Research”, and Perplexity.

The rest of this paper is organized as follows. The next section discusses previous work. Section 3 explains the Calcudoku puzzle. Our method is outlined in Section 4, followed by results in Section 5. Finally, Section 6 has conclusions and future work.

2 Previous work

This is a very active research area. To learn more, you could start with a survey paper like “A Survey on Large Language Models for Mathematical Reasoning” [Wang et al. 2025].

There are many benchmarks for math/logic reasoning in AI as well. Performance figures for many current benchmarks can be found at <https://llm-stats.com/leaderboards/best-ai-for-math>.

3 The Calcudoku puzzle

In this section we briefly describe the Calcudoku puzzle and how it is solved. This type of puzzle was invented in 2004 by a Japanese primary school math teacher named Tet-suya Miyamoto [Shortz 2009]. It is known by many names, including Calcudoku, Kashikoku Naru, Kenken, Mathdoku, Minuplu, Newdoku, etc.

Puzzles are in the form of a square grid of cells, with typical puzzles ranging in size from 4×4 to 9×9 . Given a puzzle of size $n \times n$, the solution is always a Latin square of that size, i.e. each row and each column contains each digit from the set $\{1, \dots, n\}$ exactly once. Grid cells are grouped into “cages” (groups of cells with a thick border), and marked with a clue, in the form of a result and an operator (e.g. $+$ or \times). The operator applied to the numbers in the cage must produce the result shown. The ordering of the numbers in the cage is irrelevant (as long as there exists an ordering that produces the result). The puzzle has a single solution. Evidently one needs both logic reasoning and number skills to solve this puzzle. Figure 1 shows an example simple 4×4 puzzle and its solution.

4	5+		3+
4+		4	
3+	4+		7+
	6+		

4	5+	3	3+
4	2	3	1
4+		4	
3	1	4	2
3+	4+		7+
2	3	1	4
	6+		
1	4	2	3

Figure 1: An example simple 4 × 4 Calcudoku puzzle and its solution

4 Method

4.1 Test puzzles

Puzzles are specified in the prompt in a simple ASCII format, described to the agent as follows:

I will specify a Calcudoku puzzle as follows: the first line has the size. Each next line specifies a cage like so: result, comma, operation, comma, cells. Each cell is specified by 2 characters: a letter and a number. The letter specifies the column, starting with 'a' for the left column. The number specifies the row, starting with 1 for the top row.

Given this specification, please solve the following Calcudoku puzzle:

The example puzzle of Figure 1 is specified as follows:

```
4
4,+ ,a1
5,+ ,b1c1
3,+ ,d1d2
4,+ ,a2b2
4,+ ,c2
3,+ ,a3a4
4,+ ,b3c3
7,+ ,d3d4
6,+ ,b4c4
```

Two of the puzzles had a variation on the standard puzzle: #5 was a puzzle from zero, and #6 a puzzle using the **mod** (modulo) operator. In these cases an extra explanation was added to the prompt: "Note that this puzzle uses the numbers 0-7, not 1-8" and "Note that this puzzle also uses the modulo operator %", respectively.

The test puzzles are listed in Table 1, in increasing order of difficulty. The puzzles were taken from the www.calcudoku.org website, with the listed difficulty level the same as presented on the site.

Table 1: The 7 test puzzles, in increasing order of difficulty

#	size	level	notes
1	4x4	easy	the puzzle of Fig. 1
2	6x6	easy	
3	8x8	medium	
4	6x6	difficult	
5	8x8	difficult	uses 0-7
6	8x8	difficult	uses the mod operation
7	9x9	difficult	

4.2 Tested agents

The tested agents and their models are listed in Table 2 (in alphabetical order). For agents that offer different modes (e.g. "Fast" and "Thinking"), all available modes were tested.

¹ ChatGPT itself states, when queried "Does ChatGPT have different modes, likes "fast" and "expert"?", that it does not, but that it auto-selects a model/method based on the prompt. It recommends to add:

"Solve carefully and explain every step. Double-check your result. Take your time and be precise. I want a rigorous, expert-level solution."

so this extra text was added to the ChatGPT prompt.

² Perplexity defaults to its internal "Sonar" model, and offers a wide range of other models, but only after payment. The first 5 queries use a "Pro" mode. Our tests used this "Pro" mode. One puzzle was tested using the default mode.

³ For Qwen3, the default and "Thinking" modes only apply to the "Qwen3-Max" model.

4.3 Test procedure

The puzzles were presented to each mode of each agent in increasing order of difficulty. An agent+mode would only "stay in the race" if the puzzle was solved correctly. This continued until puzzle 7, which no agent was able to solve.

The time taken was measured using an iPhone stopwatch, and is accurate ± 2 seconds.

Table 2: The tested agents, models, and modes

company and agent	model	mode(s)
OpenAI ChatGPT	GPT 5.2	default “extra prompt text” ¹
Anthropic Claude	Sonnet 4.5	default
Microsoft Copilot	GPT 5.2	Smart Think Deeper
DeepSeek DeepSeek	DeepSeek-V3.2	default DeepThink
Google Gemini	Gemini 3	Fast Thinking
xAI Grok	Grok 4.1	Fast Expert
Mistral LeChat	Mistral Large 3	default Think Deep Research
Meta Meta AI	Llama 4	Fast Thinking
Perplexity Perplexity	Sonar	default “Pro” ²
Qwen Qwen3	Qwen3-Max, Qwen3-Coder	default Thinking ³

5 Results

Table 3 lists the agents that were able to solve more than one of the 7 test puzzles. The agents are ranked first by the number of puzzles solved, then by their average solving time. The two Copilot modes are ranked “shared third” because their performance was very similar.

Table 3: Agents that solved more than 1 of the 7 test puzzles

rank	agent	# solved	avg time
1	Gemini Thinking	6	1m28
2	Grok Expert	5	2m07
3	Copilot Smart	4	1m05
3	Copilot Think Deeper	4	1m20
4	Qwen3 Thinking	4	3m08
5	DeepSeek DeepThink	4	9m01
6	Qwen3	2	2m23

Table 4 lists the agents that were only able to solve the easiest puzzle, and the time this took.

Finally, Table 5 lists the agents that were not able to solve a single puzzle, and a note from the response.

5.1 Comparison with a dedicated solver

We developed a dedicated “Calcudoku solver” program in C++. This program solved the final puzzle from the test set (that no AI agent was able to solve) in 0.13 seconds, running

Table 4: Agents that solved only the easy 4x4 Calcudoku

agent	# solved	time
LeChat Think	1	7
Perplexity Pro	1	12
LeChat	1	24
Meta Thinking	1	25
ChatGPT	1	27
Claude	1	30
DeepSeek	1	1m30
Qwen3-Coder	1	2m01

Table 5: Agents that could not solve the easy 4x4 puzzle

agent	time	note
Grok Fast	5	“unsolvable”
Gemini Fast	12	solution incorrect
Meta Fast	20	“oops no solution”
Perplexity default	22	“<x”
LeChat Deep Research	7m45	solution incorrect

in a single thread on an AMD Ryzen 7 5800X. This is much faster than the general AI agents obviously. The trade-offs between the AI and deterministic approaches are clear: the custom solver is indeed much faster, but took a lot of time and effort to develop, and can perform a single task only.

5.2 Odd answers

Table 6 shows some of the more unusual answers given by various agents.

Table 6: Unusual answers

agent	puzzle	note
ChatGPT	#2	justifies its solution by stating $1+2+3=9$
Le Chat Deep Research	#2	runs for 8m20s and fails, goes off on tangents about “CSS grids”, at some point outputs <code></think></code> repeatedly
Qwen3-Coder	#2	writes “let me just give the correct solution” after 1m37s, and produces an incorrect one
Qwen3-Max	#3	runs for almost 14 minutes and fails. After 2 minutes it still writes about “understanding the notation”
Grok Expert	#6	during the run it tried to find the puzzle on www.calculdoku.org ⁴

⁴ If Grok “Expert” had found the actual link to the puzzle (and it got close), it could have downloaded the solution from there (!)

5.3 Discussion

The top 3 agents (Gemini Thinking, Grok Expert, Copilot) produced correct results for both easy and more challenging puzzles. Gemini was especially impressive in how complex variations were handled (the puzzle “from zero” and the one with the modulo operator). Qwen3 ended up in the middle, which was somewhat surprising because it appears to perform well in other benchmarks.

ChatGPT’s claimed “automatic mode selection” did not help: it consistently struggled with even the 2nd-easiest puzzle. Microsoft’s Copilot is based on ChatGPT, but clearly outperforms it.

DeepSeek “Thinking” was slow (it took 17 minutes to solve the third puzzle, for example), but impressive in how persistent it was: it produced endless lines of output, trying many different approaches, and tracked back when unsuccessful.

It was surprising that some agents that claim to use a “coding sandbox” (e.g., LeChat Think, Qwen3-Coder) did poorly. The fact that the “Fast” agents didn’t do well is not a surprise.

Poor performers overall were ChatGPT, Claude, LeChat, Meta AI, and Perplexity.

6 Conclusions and future work

In summary, current AI agents (when not in “pure text mode”) are already surprisingly good at solving difficult Calculdoku puzzles. Our custom-made deterministic program is still much faster, but it took a *long* time to develop, and obviously can do one thing only.

The Google Gemini “Thinking” and xAI Grok “Expert” modes were especially impressive. The results correlate somewhat with other AI math/reasoning benchmarks, but are difficult to compare (other benchmarks include non-free models for example). The field of AI benchmarking is very much in flux as well.

In future work, we would like to:

- expand the test set so it includes more puzzles that the top performers fail on, and make the difficulty level increases more gradually
- publish the test set so it can become a benchmark, *or* perhaps publish a similar “training set” and keep the test set secret
- runs tests at different times of day to see if this impacts performance
- test models and modes that we may have missed
- compare the results to similar benchmarks

Your comments and suggestions are very welcome at calculdoku@gmail.com.

References

- SHORTZ, W., 2009. A new puzzle challenges math skills. www.nytimes.com/2009/02/09/arts/09ken.html.
- WANG, P.-Y., LIU, T.-S., WANG, C., LI, Z., WANG, Y., YAN, S., JIA, C., LIU, X.-H., CHEN, X., XU, J., AND YU, Y. 2025. A survey on large language models for mathematical reasoning. *ACM Computing Surveys* (December).