

830
831
832
833
834
835

Supplementary Materials for Creativity or Brute Force? Using Brainteasers as a Window into the Problem-Solving Abilities of Large Language Models

836	A Limitations	21
837	B Broader Impact	21
838	C Full Dataset Statistics	21
839	D API Inference Settings	21
840	E Prompts	22
841	F Counting Steps	22
842	G Braingle Math Dataset Categorization	29
843	G.1 Overview	29
844	G.2 Examples	30
845	H Braingle Logic Dataset Categorization	31
846	H.1 Overview	31
847	H.2 Examples	32
848	H.3 Full Population Statistics	37
849	I Model Performance by Categories	39
850	J Brute-Force	41
851	J.1 Brute Force Prompting Analysis	41
852	J.2 Brute Force Statistics	42
853	J.3 Brute Force by Problem Category	42
854	J.4 Correlation of Solution Summarization Ability with Correctness/Brute-Force Usage	45
855	K Case Study for Informed Self-Correction	47
856	K.1 Informed Self-Correction prompt	47
857	K.2 Informed Self-Correction Results	47
858	K.3 Error Misattribution	47
859	K.4 False Confession	48
860	K.5 Justified Denial	49
861	K.6 Over-Elaborated Response	51
862	L Recurring Error Patterns in OpenAI o3 and Prompt-Based Remedies	53

863 A Limitations

864 Our study’s scope has several limitations. The effectiveness of prompt engineering was constrained,
 865 as simple discouragement of brute-force had minimal impact, and even hints showed varied success.
 866 Our findings are specific to the BRAINTEASER benchmark, derived mainly from one source and
 867 focused on math/logic puzzles, which may limit generalizability. The use of OpenAI o3 for certain
 868 evaluations and our macro-level analysis of solution step counts could introduce biases or mask
 869 detailed computational efforts.

870 Beyond these, LLMs frequently defaulted to brute-force over creative insight, faced challenges in
 871 robust self-correction (including "false confessions"), and didn’t consistently improve when puzzles
 872 were formalized mathematically. These observations, from a specific set of LLMs and our operational
 873 definitions of creativity and brute-force, highlight current boundaries in advanced LLM reasoning.

874 B Broader Impact

875 This work provides a systematic framework for probing the reasoning capabilities of large language
 876 models (LLMs) through the lens of brainteasers, offering insights into how these systems approach
 877 complex problem-solving tasks. By distinguishing between creative and brute-force strategies, the
 878 BRAINTEASER benchmark encourages the development of models that reason more efficiently and
 879 human-like, rather than relying on computational brute force. This shift toward more nuanced
 880 reasoning abilities could enhance AI’s applicability in fields requiring critical thinking, such
 881 as education, scientific discovery, and collaborative problem-solving, where transparency and
 882 interpretability of reasoning processes are essential.

883 C Full Dataset Statistics

Table S1: Population statistics of the datasets curated.

Statistics↓ Data→	Braingle Math Most difficult ($n = 250$)	Braingle Logic Most difficult ($n = 250$)
Percentage with hints	100%	100%
Difficulty score out of 4 median	2.80 ± 0.15 2.79	2.66 ± 0.23 2.64
Answer word count median	172 ± 204 109.5	237 ± 236 162.5
Answer sentence count median	7.01 ± 8.13 4.00	10.88 ± 10.91 8.00

884 D API Inference Settings

Model	Top _p	Max Tokens	Temperature
DeepSeek R1 Distill Qwen 1.5B	0.7	10000	0.7
DeepSeek R1 Distill Qwen 14B	0.7	10000	0.7
DeepSeek R1 Distill Llama 70B	0.7	10000	0.7
deepseek-chat (Deepseek-V3)	1	10000	1
deepseek-reasoner (Deepseek-R1)	-	10000	1
gemini-2.5-flash-preview-04-17	0.95	10000	1
OpenAI o3	1	10000	1

885 E Prompts

Prompt: Chain of Thought (CoT)

You are an excellent mathematician with perfect logic. You are also very patient, and willing to perform very long chains of reasoning when necessary.
Can you solve this problem? Please spell out your entire reasoning steps. Finish your response in the format of “Final answer: ” immediately followed by your answer.

886

Prompt: Math Prompt

You are an excellent mathematician with perfect logic. You are also very patient, and willing to perform very long chains of reasoning when necessary. Solve the given problem, keeping in mind the following:
If you use a brute force or guess-and-check method or utilize code when not necessary, you will receive no credit. If you do not fully justify a step, you will receive no credit.
Ensure each of your statements is consistent with the conditions of the problem and statements you have already written before moving on. If you do not do this, you will receive no credit.
An outline of a solution without a concrete final answer will also receive no credit.
The problem statement is correct, and a correct answer exists. If you solve a version of the problem or attempt to modify the problem statement to a version other than the one written exactly as-is, you will receive no credit.
Finish your response in the format of “Final answer: ” immediately followed by your answer.

887

Prompt: Hint

You are an excellent mathematician with perfect logic. You are also very patient, and willing to perform very long chains of reasoning when necessary.
Can you solve this problem using the hint? Please spell out your entire reasoning steps. Output the final answer in the format of “Final answer: ” at the end of your answer.

888

889 F Counting Steps

890 To have OpenAI o3 and DeepSeek R1 interpret solutions, break them down into steps, and label steps
891 as creative or rudimentary, we presented the following prompt.

Prompt: Counting Steps

I will provide you with a problem and its solution. Without solving the problem yourself, I want you to divide the provided solution into steps. Then, I want you to distinguish each step as either a creative step or a rudimentary step. A creative step is a step that generates innovative insights that reduce the problem or make the problem significantly easier to solve. Common traits of creative steps include using analogies, combining ideas from different domains, exploiting problem constraints, or devising elegant, efficient strategies that go beyond straightforward computation or trial-and-error. A rudimentary step is a step that applies creative insights and often is more easily or routinely derivable, such as straightforward computation; it could also be a step that may make progress in solving the problem but is not innovative, such as using trial-and-error or systematically exploring all possible options. Common traits of rudimentary steps include utilizing code, guess-and-check, or performing computations that a human would ordinarily not be able to do. In terms of defining a step, note that a step is defined as something that does not need to be super fine-grained like " $1 + 1 = 2$ " or " $a \rightarrow b, b \rightarrow c; a \rightarrow c$." Instead, each step should represent a key component of the solution and the steps sequentially lead to the final answer to form a complete solution. Make sure to not omit any necessary information, and also make sure that your standard of what is considered a "step" is as consistent as possible. Count the total number of steps in the human solution and report each step and the total number of steps in the format "Total Step Count:" followed by the number of steps. Also, keep count of the number of creative steps and rudimentary steps. After displaying the total step count, please report "Creative Steps:" followed by the number of creative steps and "Rudimentary Steps:" followed by the number of rudimentary steps. Lastly, report the steps themselves in the format "Steps:" with the steps needed to solve the problem.

892

893 As mentioned in section 3.2, models are often able to correctly comprehend the idea that a step being
894 a key component of the problem does not necessarily imply that every individual deduction is a step.
895 We now present explicit examples of steps that consist of multiple deductions which combine to a
896 more key component of the problem. For example, for math problems that require choosing between
897 several candidates to deduce mystery numbers, the models consider the analysis of all candidates as
898 one collective step rather than dedicating one step for analyzing each candidate. Take the following
899 output from OpenAI o3.

Model Response: Math 10 (Output Step Example)

For each of those products, ask whether Prashant could become certain after hearing Sachin's opening remark. Only product 52 qualifies, because every other product can be factored in more than one way that keeps the sum inside or outside the Possible-Sum list.

900

901 Notice that multiple computations would be required in analyzing each individual candidate, let alone
902 all candidates; even so, the model is able to combine these rudimentary, brute force-like calculations
903 as an overall step, in turn adhering to our intended definition of a step of representing a substantial
904 amount of progress to solving the problem rather than every individual deduction.

905 For logic puzzles with a given list of clues, the use of a clue is considered a single step, even when
906 multiple strings of thought are made, which is, again, as intended.

Model Response: Logic 6 (Output Step Example)

Use the "only one ascending triple" condition (clue 7).
• Exactly 1 lower than Q can be left of her (or 2 triples would appear), triple must be Q-?-A.
• To prevent J-K-A or T-K-A from creating a second triple, K must be to the far left.
• To avoid T-J-A making another triple, J must precede T.
Complete rank order: K J Q T A.

907

908 Now, we consider the following problem that wants to arrange two 1's, 3's, 7's, and 9's in a string
909 such that each pair of consecutive integers is prime, as well as the number itself and its reverse.
910 We will then compare the outputs of OpenAI o3 and DeepSeek R1 to observe any similarities and

911 differences in how solutions are divided into steps and how steps are categorized as either creative or
912 rudimentary.

Problem: Math 21

Arrange the numerals "11337799" to form an 8-digit number (referred to as "ABCDEFGH") that meets the following conditions:

- 1) ABCDEFGH is a prime number
- 2) HGFEDCBA (reversing the digits) is a prime number
- 3) all consecutive 2-digit subsets of ABCDEFGH are prime numbers (AB, BC, CD, DE, EF, FG, GH are all prime)
- 4) all consecutive 2-digit subsets of HGFEDCBA are prime numbers (HG, GF, FE, ED, DC, CB, BA are all prime)

913

914 Notice that the creative step uses a mathematical insight regarding reversible primes containing 9
915 to reduce the problem into more limited space of possibilities, while the rudimentary step simply
916 uses trial-and-error on all possibilities of the remaining digits to find valid combinations. Again, the
917 explicit steps themselves are provided below.

Human Solution: Math 21

The only available 2-digit prime numbers are: 11, 13, 17, 19, 31, 37, 71, 73, 79, and 97. The number pairs that are prime in both directions are: 11/11, 13/31, 17/71, 37/73, and 79/97 (19 is eliminated since its reverse, 91, is not prime). With these numbers, it can be seen that 9 can only be paired with 7. Therefore, the solution must be of the form 97????79, 9797????, or ???7979. ??? must therefore be 1313, 3131, or 3113. This leaves the possible solution pairs of 97131379/97313179, 97971313/31317979, 97973131/13137979, 97973113/31137979, and 97311379/97311379 (the last pair being a palindrome). Of these, only 97131379 and its reverse, 97313179, are prime numbers.

918

Model Response: Math 21 (OpenAI o3)

1. List every two-digit prime that can be built from the available digits 1, 3, 7, 9: 11, 13, 17, 19, 31, 37, 71, 73, 79, 97. (Rudimentary)
2. From that list, keep only those whose reversals are also prime, because each two-digit block must work in both the forward and the reverse 8-digit number. The viable pairs are 11/11, 13/31, 17/71, 37/73, 79/97. (Rudimentary)
3. Observe that the digit 9 can appear in a two-digit prime only as 79 or 97, so every 9 must be adjacent to a 7. (Creative)
4. Use the previous observation to pin down the overall skeleton of the 8-digit number: it must be of the form 97 ____ 79, 9797 ____, or ____ 7979. (Creative)
5. Insert the remaining digits 1,1,3,3 into the four open slots so that every new two-digit block is still in the list from Step 2, leaving only three feasible fillings: 1313, 3131, or 3113. (Rudimentary)
6. Combine the frameworks from Step 4 with the fillings from Step 5 to generate the complete candidate pairs of 8-digit numbers (each with its reverse): 97131379 / 97313179, 97971313 / 31317979, 97973131 / 13137979, 97973113 / 31137979, 97311379 / 97311379. (Rudimentary)
7. Test each candidate for primality; only 97131379 and its reverse 97313179 are prime, so they satisfy all conditions. (Rudimentary)

919

Model Response: Math 21 (DeepSeek R1)

1. List all possible 2-digit primes from the given digits (11, 13, 17, 19, 31, 37, 71, 73, 79, 97). (Rudimentary: This is a straightforward enumeration of primes from the given digits.)
2. Eliminate primes whose reverses are not prime (e.g., 19 is eliminated because 91 is not prime). (Creative: This insight reduces the problem by filtering out invalid candidates early.)
3. Identify digit constraints (e.g., 9 can only be paired with 7). (Creative: This observation narrows down the possible starting and ending digits of the solution.)
4. Determine possible forms of the solution (97????79, 9797????, or ?????7979). (Creative: This step leverages the digit constraints to structure the problem into manageable subproblems.)
5. Enumerate possible middle segments (1313, 3131, or 3113). (Rudimentary: This is a systematic exploration of possible combinations.)
6. Generate all candidate pairs based on the forms and middle segments. (Rudimentary: This is a straightforward generation of possible numbers from the constraints.)
7. Check which candidates and their reverses are prime numbers. (Rudimentary: This involves computational verification of primality.)
8. Conclude that only 97131379 and its reverse 97313179 satisfy all conditions. (Creative: This final step synthesizes the results and confirms the unique solution.)

920
921 Notice that both models divide the steps nearly identically, with the only difference being that the
922 DeepSeek model divides the testing of final candidates and the declaration of the final answer as two
923 steps whereas OpenAI o3 only considers this as one step. More importantly, both models' nearly
924 identical breakdowns also correctly follow the human solution's logic in the exact sequence it was
925 presented; this exemplifies the models' abilities to properly parse a provided response into steps
926 while maintaining its overall logical structure. Now, we observe both models' steps 3, 4, and 5. Both
927 models declare steps 3 and 4 as creative in making more sophisticated deductions on prime numbers
928 including 9 and significantly narrowing down the possible forms of the solution, which correspond to
929 our definition of creative steps in making an innovative insight without systematically exploring all
930 possible options. Contrarily, both models correctly declare the brute-force exploration of feasible
931 segments of 1's and 3's as a rudimentary step.

Table S2: Mean and standard deviation of solution step counts for several models' (OpenAI o3, DeepSeek V1, Llama 70B) solutions for the most difficult puzzles in the Braingle Math and Braingle Logic datasets (n=250) based on OpenAI o3. DeepSeek V1 solutions were used instead of DeepSeek R1 solutions because too many DeepSeek R1 responses only consisted of the final answer without the reasoning process that we can divide into steps.

	Solution Steps	Median	Creative Steps	Rudimentary Steps
Braingle Math				
OpenAI o3 Solution	8.2 ± 3.5	8.0	1.9 ± 1.2	6.3 ± 3.2
DeepSeek V1 Solution	11.2 ± 4.6	10.0	2.9 ± 2.5	8.3 ± 3.7
Llama 70B Solution	9.6 ± 3.4	9.0	2.6 ± 1.8	7.0 ± 3.2
Braingle Logic				
OpenAI o3 Solution	8.1 ± 3.8	7.0	2.2 ± 1.8	5.8 ± 3.2
DeepSeek V1 Solution	12.3 ± 5.2	11.0	3.3 ± 2.5	9.0 ± 4.6
Llama 70B Solution	11.0 ± 4.6	10.0	2.9 ± 2.1	8.1 ± 4.3

932 However, as noted in Table S2 and S3, we notice that DeepSeek R1 tends to declare more steps
933 as creative than OpenAI o3, and we notice that here, DeepSeek declares steps 2 and 8 as creative
934 while OpenAI o3 considers them to be rudimentary. By our definitions, we expect this to be more
935 rudimentary; step 2 requires scanning all of the prime numbers to see which primes have prime
936 reversals, which would be rudimentary, but DeepSeek considered this creative for filtering invalid
937 candidates. Despite the significant progress, the greater focus should have been placed on the
938 systematic exploration of the option space. This may be an issue with prompting, or could be the
939 fact that DeepSeek may be associating significant progress with creativity rather than the innovation
940 of the ideas used. Then, step 8 from DeepSeek is declaring the final answer that should have been
941 obtained from step 7, so this should be the most rudimentary because no additional deduction was
942 made. It is interesting to see that DeepSeek R1 considers this as creative because it synthesizes the
943 results, but after observing other DeepSeek responses, we see that calling the declaration of the final
944 result a creative step is not a common error and that erroneously associating significant progress

Table S3: Mean and standard deviation of solution step counts for the 30 most difficult puzzles in the Braingle Math and Braingle Logic datasets. (*For math, 1 human solution and 1 model solution were excluded as outliers due to excessive casework, 1 human solution excluded due to directly asserting answer)

	Solution Steps	Median	Creative Steps	Rudimentary Steps
Braingle Math – OpenAI o3 Step Count				
Human Solution ($n = 28$)*	7.8 ± 3.5	7.0	2.1 ± 1.4	5.7 ± 2.6
OpenAI o3 Solution ($n = 29$)*	7.7 ± 2.9	8.0	2.0 ± 1.0	5.7 ± 3.0
OpenAI o3 Correct Solution ($n = 21$)	8.0 ± 2.2	8.0	1.9 ± 0.9	6.1 ± 2.4
Braingle Math – Deepseek R1 Step Count				
Human Solution ($n = 28$)*	7.8 ± 4.7	7.0	3.9 ± 2.3	3.9 ± 2.7
OpenAI o3 Solution ($n = 29$)*	6.8 ± 3.2	6.5	3.3 ± 1.3	3.5 ± 2.5
OpenAI o3 Correct Solution ($n = 21$)	7.1 ± 3.1	7.0	3.2 ± 1.3	4.0 ± 2.5
Braingle Logic – OpenAI o3 Step Count				
Human Solution ($n = 30$)	10.6 ± 5.2	9.0	2.9 ± 1.7	7.8 ± 5.5
OpenAI o3 Solution ($n = 30$)	N/A	N/A	N/A	N/A
OpenAI o3 Correct Solution ($n = ??$)	N/A	N/A	N/A	N/A
Braingle Logic – Deepseek R1 Step Count				
Human Solution ($n = 30$)	9.6 ± 4.7	9.0	5.3 ± 2.3	4.3 ± 2.6
OpenAI o3 Solution ($n = 30$)	-	-	-	-
OpenAI o3 Correct Solution ($n = ??$)	-	-	-	-

rather than innovation with creativity may be a more likely source of why DeepSeek R1 categorizes a greater number of steps as creative than OpenAI o3.

From Table S2, we observe that DeepSeek V1 solutions have more steps than Llama 70B solutions for both math and logic. OpenAI o3 outputs have fewer steps than both of the other models for both math and logic brainteasers. We also notice that logic problems generally have more steps than math problems as seen with DeepSeek V1 and Llama 70B solutions; that said, for OpenAI o3, the average step count for logic problems is marginally smaller. The DeepSeek V1 and Llama 70B responses are both considerably longer than human solutions, which may be a result of continued trial-and-error attempts that would not be noted in human solutions.

We now present an example of an incorrect solution from DeepSeek V1 followed by the OpenAI o3 step breakdown. As the DeepSeek solution is very long, some portions are truncated but the overall sequence of logic is still present.

Model Response: Math 11 (DeepSeek V1)

First, let's clarify some terms to ensure we're on the same page.

Understanding the Terms

1. **Palindrome:** A palindrome is a number (or word) that reads the same backward as forward. For example, 121, 1331, and 1 are palindromic numbers.
2. **Non-palindromic integer:** An integer that is not a palindrome. For example, 12, 123, and 10 are non-palindromic, since reading them backward gives 21, 321, and 01, respectively.
3. **Cube of an integer:** The integer multiplied by itself three times. For example, the cube of 3 is

$$3^3 = 3 \times 3 \times 3 = 27.$$

The Problem Restated

We're looking for an integer n such that n is *not* a palindrome, n^3 is a palindrome, and n is the only such integer.

Checking Small Integers

1. $n = 1$:
 - 1 is a palindrome, $1^3 = 1$, which is a palindrome, excluded since n itself is palindromic.
2. $n = 2$:
 - 2 is a palindrome, $2^3 = 8$, which is a palindrome, excluded since n is palindromic.
- \vdots
3. $n = 26$:
 - 26 is not a palindrome, $26^3 = 17576$, which (by the problem's assertion) is a palindrome, this is the unique solution.

Considering Negative Integers

A negative integer is non-palindromic if its positive counterpart is not a palindrome (since the negative sign makes it non-palindromic, as palindromes are typically considered for absolute values in numbers).

Let's check some negative integers. (The model then checks -1, -2, -10, and -11, but gives up after finding no success).

Perhaps the answer is among positive integers, and I missed it earlier.

Re-evaluating Positive Integers

Let me think about the cube being a palindrome. The cube must read the same backward as forward.

Looking back, the only cubes that are palindromes from the numbers I checked are 1 (from 1), 8 (from 2), 343 (from 7), and 1331 (from 11).

But all these ' n ' values are palindromic themselves.

Wait, is there any non-palindromic ' n ' whose cube is a palindrome? Maybe higher numbers? (The model then tries 101 and 1001 but fails as 101 and 1001 are palindromes).

Model Response: Math 11 (DeepSeek V1) continued

This is perplexing. Maybe the answer is that there is no such non-palindromic integer whose cube is a palindrome. But the problem states that there is one, implying it exists.

Wait, perhaps I missed a small number. Let me check ' $n = 0$ '. 0 is a palindrome, $0^3 = 0$, which is a palindrome, but 0 is palindromic.

Alternatively, maybe the answer is 10, but earlier we saw $10^3 = 1000$ isn't a palindrome. Unless we consider 1000 as a palindrome if we ignore trailing zeros, but that's not standard.

After some research or recalling, I think the answer is ****26****, but earlier calculation showed $26^3 = 17576$, which isn't a palindrome. Maybe I miscalculated.

Let me calculate 26^3 carefully. (The model obtains that $26^3 = 17576$. 17576 backward is $67571 \neq 17576$. So not a palindrome.)

(The model then continues to try numbers it has already tried out, such as 0, 10, -1, and 26, but still is unsuccessful.)

:

Given that, and since the problem states there is one, the most likely answer is ****26****, accepting that 17576 is a palindrome (perhaps by some definition where the middle digit is the center and the others mirror, but they don't here).

However, standard definition says a palindrome reads the same backward, which 17576 does not ($67571 \neq 17576$).

Given this confusion, and based on mathematical sources, the accepted answer is:

****26****, with $26^3 = 17576$ being considered a palindrome, even though by strict digit reversal it's not. This might be a special case or an exception in how palindromes are defined for this problem.

958

Model Response: Math 11 (OpenAI o3 Breakdown)

1. State and explain the relevant definitions of "palindrome," "non-palindromic integer," and "cube," establishing the criteria the sought integer must satisfy. (Rudimentary)
2. Propose a brute-force method: examine successive integers, compute each cube, and check both n and n^3 for palindromicity. (Rudimentary)
3. Implement the method for $n = 1-9$; record that either n is itself a palindrome or n^3 is not, so none of these values fits the requirement. (Rudimentary)
4. Continue the same manual search for $n = 10-22$ and again find no qualifying integer. (Rudimentary)
5. Rely on a recollection/intuition and single out $n = 26$ as a promising candidate instead of continuing the systematic scan in order. (Creative)
6. Calculate $26^3 = 17576$ and test 17576 for palindromicity, concluding (incorrectly) that it is a palindrome. (Rudimentary)
7. Declare that, because 26 appears to work and no other candidates have been discovered, 26 must be the unique non-palindromic integer whose cube is a palindrome, and give 26 as the final answer. (Rudimentary)

959

960 From reading the outputs, it is clear that OpenAI o3 maintains the same chain of thought as the
961 DeepSeek V1 response while consolidating much of the trial and error performed on previously
962 tested candidates. Like DeepSeek V1, the OpenAI o3 also notes that 17576 is not a palindrome. We
963 also notice that the steps are larger components of the problem rather than just individual deductions
964 as desired, though we could argue that steps 3 and 4 could be combined since the method of search
965 is identical in both steps. Nonetheless, we still see that OpenAI o3 breaks down a provided model

Table S4: An example of creative and rudimentary steps.

Creative Step Example	Rudimentary Step Example
Examine the given input-output pairs and discover the hidden rule: write each number in binary and weight each digit—count 1 for every 0 and 2 for every 1.	Convert 9304 to binary, obtaining 10010001011000.

response into macro-steps that correctly highlight the chain of thought even when the response can yield an incorrect answer. This shows that the OpenAI o3 step breakdowns do not look beyond the provided response itself and maintain the thought process given by the response.

One drawback of using the step count to measure the complexity of a problem is that the complexity may vary significantly between steps. For many problems, there are fewer insights, but the insights are more difficult to obtain and are more significant in solving the problem; yet the model’s presentation of these steps does not necessarily highlight the differences in complexity between steps. This is particularly true for math problems that ask for some common pattern between a list of numbers without giving any hint of the pattern. Without acknowledging the required trial-and-error and time needed to discover the step’s key insight, such “steps” would be considered equivalent in complexity to routine arithmetic calculations under this measurement. Therefore, we also make the distinction between creative steps and rudimentary steps. In the problem that provides a set of input-output pairs where the goal is to find the corresponding output of 9304, we show in Table S4 that finding the input-output rule is considered creative as it significantly reduces the difficulty of the problem, while carrying out the rule on converting the designated number 9304 is rudimentary since calculations like binary conversion are simple and routine. The explicit steps themselves are provided in Supplementary Section F.

G Braingle Math Dataset Categorization

G.1 Overview

G.1.1 Categories

We manually categorize the math dataset based on problem style, into the following three categories and respective subcategories. Category (1) represents problems which could reasonably be found on a standard math competition such as AMC, AIME, etc. Category (2) represents problems which are still mathematically rigorous, but do not quite fit the style of a competitive math problem. Category (3) represents problems which are not rigorous, and involve finding patterns or thinking outside the box.

1. Standard competitive math:

- **Geometry** - Problem with spatial reasoning, finding areas/side lengths, etc.
- **Number Theory** - Divisibility, factorization, etc.
- **Combinatorics** - Counting, probability, etc.
- **Algebra** - Setting up and solving systems of real-valued equations

2. Nonstandard:

- **Logic** - Problems resembling mathematical Logic puzzles
- **Special Number** - Finding a “mystery” number with desired properties

3. Heuristic/non-rigorous:

- **Pattern** - Finding the “next number” or a common rule for a group or sequence of numbers
- **Arithmetic** - Rearranging an arithmetic expression to achieve a target number

1004 **G.2 Examples**

1005 **G.2.1 Standard category**

Problem: Math 50 (Geometry)

It's easy to see that a ring can completely hold (surround) two identical smaller rings with half the diameter, without overlapping. Three times the diameter, the bigger ring is space enough to seat seven rings; the outer six touching both the middle ring and the bigger circle/perimeter. Using this basic information and your imagination, determine the maximum number of rings that could be housed inside another ring with four, five, six and seven times the diameter.

1006

1007 Geometry problems require spatial reasoning– in this case, the solver must visualize an arrangement
1008 of circles contained within a larger circle, and argue about the “total diameter” of this arrangement.

Problem: Math 124 (Number Theory)

I have a machine which has four cog wheels in constant mesh. The largest cog has 242 teeth and the others have 160, 64 and 22 respectively. How many revolutions must the largest cog make before each of the cogs is back in its starting position?

1009

1010 Number theory problems often involve concepts like common divisors, multiples, and remainders.

Problem: Math 117 (Combinatorics)

What is the largest number of pieces you can form with n straight cuts of a pizza? The pieces do not need to be of equal size.

1011

1012 While this may seem like a problem requiring a visual arrangement, a clever counting argument can
1013 simplify the problem greatly: for every cut added, if it intersects k of the existing cuts, it adds $k + 1$
1014 new pieces to the arrangement.

Problem: Math 125 (Algebra)

Three people (A, B, and C) need to cross a bridge. A can cross the bridge in 10 minutes, B can cross in 5 minutes, and C can cross in 2 minutes. There is also a bicycle available and any person can cross the bridge in 1 minute with the bicycle. What is the shortest time that all men can get across the bridge? Each man travels at their own constant rate.

1015

1016 Algebra problems often involve word problems with times, prices, and rates; they can be easily
1017 computed after a proper mathematical setup from the word problem.

1018 **G.2.2 Nonstandard category**

Problem: Math 122 (Logic)

The people of Olde Mathville had unique ways of punishing wayward citizens. For example, those convicted of crimes of dishonesty were chained to the Liars' Rail until they solved a number of puzzles.

One such puzzle has been recently discovered! In the multiplication below, each letter - L, I, A, R, and S - takes the place of a different digit. Find the digits to make the multiplication true.

```

L I A R
x  S
-----
R A I L

```

1019

1020 While this problem is still mathematically themed, the primary focus of the problem is in logical
1021 reasoning– the solver must assign five unique values to the five unknown entities, as one would in a
1022 logic grid.

Problem: Math 75 (Special Number)

What is the smallest number that is the sum of two different pairs of cubes?

1023
1024 In this category, the solver must find a number or set of numbers satisfying a unique set of constraints.
1025 This category can be seen as adjacent to **Number Theory**; however, in **Special Number** puzzles, the
1026 primary challenge is not in making rigorous number-theoretic arguments relating to divisibility, but
1027 rather deductive, heuristic, and sometimes “lucky” observations.

1028 **G.2.3 Heuristic category**

Problem: Math 79 (Pattern)

What is so special about this particular sequence of numbers?
425260376469080434957

1029
1030 Here, the problem does not have a rigorous final answer, but rather requires general pattern-finding.
1031 In this problem, the listed numbers are the digits of pi, with 1 added to each. We review all such
1032 problems to ensure that the final answer is both “reasonable” and the unique “reasonable” solution
1033 (i.e. no other plausible patterns exist).

Problem: Math 216 (Arithmetic)

Add the appropriate mathematical operators or symbols to make the following correct:
3 4 5 = 90
Parentheses may be used freely.

1034
In these problems, the solver inserts mathematical operators to achieve a desired numerical goal.
We place these kinds of problems in the **Heuristic** category, as the solver is often required to think
outside the box, and the set of valid “actions” is not always rigorously defined. In this case, the
answer requires use of a factorial:

$$3/4 \times 5! = 90$$

1035 **H Braingle Logic Dataset Categorization**

1036 **H.1 Overview**

1037 **H.1.1 Metrics**

1038 For each brainteaser, where applicable, we evaluate the following metrics:

- 1039 • **Depth:** The number of distinct traits in each “grouping”
- 1040 • **Width:** The number of “groups”
- 1041 • **State Space Size:** The total number of possible arrangements within the initial defined constraints
1042 of the problem
- 1043 • **Number of clues:** The number of distinct pieces of information given, AFTER the state space is
1044 defined

1045 **H.1.2 Categories**

1046 We manually categorize the logic dataset based on general structure of each problem, into the
1047 following four categories and respective subcategories. Note that categories (1) and (2) represent
1048 rigorous logical deduction problems with bounded state spaces. In (3), problems do NOT have well-
1049 defined or bounded state spaces, with answers ranging from numbers to descriptions of algorithms.
1050 In (4), answers are not logically rigorous, but can still reflect a model’s pattern-finding or heuristic
1051 reasoning skills.

- 1052 • **(1) Large state space, simple clues:**
- 1053 – **0D Logic Grid** - Standard logic grid with no positional reasoning. d traits, w options per trait
- 1054 – **1D Positional** - Puzzle with 1D positional reasoning. Includes races, seating arrangements, etc.
- 1055 – **2D Positional** - Puzzle with 2D positional reasoning. Includes Bingo cards, chess boards, etc.
- 1056 – **Number** - Guess the number; deduction-style problem with mathematical clues. $d = 2$, w digits
- 1057 – **Clusters** - No fixed traits, cluster wd objects freely into w groups of d
- 1058 – **Tree** - Puzzle with tree-like structure. Includes family trees, elimination-style tournaments.
- 1059 • **(2) Small state space, complex clues:**
- 1060 – **Liars** - Clues themselves may be true or false
- 1061 – **Communication** - Impartial information between multiple parties, typically through conversation
- 1062 – **Compound** - Multi-claused or conditional clues (i.e. many if, then, or statements)
- 1063 • **(3) Math-like:**
- 1064 – **Algorithm** - Design and/or execute an algorithm
- 1065 – **Math** - Similar to comp-math, unbounded numerical state space
- 1066 • **(4) Heuristic/non-rigorous:**
- 1067 – **Pattern** - Observing a non-rigorous pattern, “riddle”-like problems
- 1068 – **Linguistic** - Clues based on semantic meaning

1069 H.2 Examples

1070 H.2.1 Computing metrics

1071 A statement defining “state space” is a statement which gives general information about the problem,
 1072 which is standard among other problems of the same type. This may include defining traits (name,
 1073 occupation, age), defining options per trait (“occupations are welder, bricklayer, ...”), setting number
 1074 of items per category (“there are 2 welders, 1 bricklayer...”), and declaring other constraints (no
 1075 repeating digits). In contrast, a “clue” is defined as a piece of information which is unique to the given
 1076 problem. Note that for some problems, the concept of “depth” and “width” can be up to interpretation.
 1077 These metrics are made to generally quantify the difference in the various categories.

1078 Here is an example solution for a classic logic grid-style puzzle:

Problem: Logic 15

1st House: Yellow, Norwegian, Water, Cats, Dunhill
 2nd House: Blue, Dane, Tea, Horse, Blends
 3rd House: Red, Brit, Milk, Birds, Pall Malls
 4th House: Green, German, Coffee, FISH, Prince
 5th House: White, Swede, Beer, Dogs, Bluemasters

1079
 1080 Each “grouping” contains 6 traits: house, color, nationality, beverage, pet, and cigar, so the **depth** is
 1081 6. There are 5 groupings, so the **width** is 5. For a classic logic grid with depth d and width w , the
 1082 **state space** is given by: $(w!)^{d-1} = (5!)^{6-1}$. Note that we define state space as the entire required
 1083 arrangement, NOT just the space of possible final answers.

Problem: Logic 3

There is a ten-digit mystery number (no leading 0), represented by ABCDEFGHIJ, where
 each numeral, 0 through 9, is used once. Given the following clues, what is the number?

- 1) $A + B + C + D + E$ is a multiple of 6.
- 2) $F + G + H + I + J$ is a multiple of 5.
- 3) $A + C + E + G + I$ is a multiple of 9.
- 4) $B + D + F + H + J$ is a multiple of 2.
- 5) AB is a multiple of 3.
- 6) CD is a multiple of 4.
- 7) EF is a multiple of 7.
- 8) GH is a multiple of 8.
- 9) IJ is a multiple of 10.
- 10) FE , HC , and JA are all prime numbers.

1085 Here, a “grouping” is simply pairing digit positions with digits, giving a **depth** of 2. There are ten
 1086 positions, giving a **width** of 10. There are ten **clues** (not counting the initial set up information), and
 1087 the **state space size** is given by $9 \cdot 9!$, accounting for the non-leading zero information.

1088 H.2.2 Large/simple category

Problem: Logic 32 (0D)

These five businessmen represented different companies at a recent trade fair. Unfortunately, the hotel they were all due to stay in had accidentally double booked their rooms. They tried other hotels in the area, but all were fully booked. Consequently, they all agreed to share the only two rooms available in the hotel - one twin and one triple. From the clues, can you work out each man’s name, company and official title?

1. When the five men realized their dilemma, they drew straws to see who would share with whom. The outcome for four of them was that the CEO shared with the businessman from ABM Inc. and Edgar shared with the Developer.
 2. Alan does not work for Reed Right and is not the Director. The Director does not work for Lantel or Blue Teeth.
 3. Neither Clarkson nor Grimaldi works for Reed Right. Grimaldi is either the IT Analyst or the CEO.
 4. The businessman from Reed Right did not share a room with Thomas.
 5. Clarkson, who does not work for Lantel, shared with either Thomas or Grimaldi but not both.
 6. Edgar’s surname is either Casson or Graves. Edgar did not share with Carl.
 7. The businessman from Chiptech shared with the businessman from Lantel.
 8. The CEO shared with the IT Analyst.
 9. Casson works for either Blue Teeth or Chiptech.
 10. Neither Joshua nor Alan works for ABM Inc.
 11. Fielder shared with the Director.
 12. The Accountant from Blue Teeth shared with the Developer.
- First names: Alan, Carl, Edgar, Joshua, Thomas.
 Last names: Casson, Clarkson, Fielder, Graves, Grimaldi.
 Company: ABM Inc., Blue Teeth, Chiptech, Lantel, Reed

1089
 1090 While there is some variation, the majority of puzzles in the 0D category follow the standard logic
 1091 grid format, where we are given a fixed set of categories (first names, last names, companies), and a
 1092 set of clues which each give a fairly simple individual deduction.

Problem: Logic 178 (1D)

List the order in which each person finished.
 Tommy Tombstone finished after Lance Lamers and Brett Brown but before Mitch Monday.
 Peter Poultry finished before Daniel Dusk and Lance Lamers.
 Sam Sunny finished after Peter Poultry and before Jack Jill and Harry Hills.
 Keri Kernel finished after Peter Poultry, Mitch Monday and Tommy Tombstone.
 Lance Lamers finished after Brett Brown and Daniel Dusk, but before Jack Jill and Mitch Monday.
 Mitch Monday finished after Sam Sunny and Brett Brown.
 Brett Brown finished before Jack Jill, Mitch Monday and Peter Poultry.
 Daniel Dusk finished before Keri Kernel and Tommy Tombstone, but after Sam Sunny.
 Jack Jill finished before Keri Kernel, Tommy Tombstone and Mitch Monday, but after Peter Poultry and Daniel Dusk.
 Harry Hills finished before Mitch Monday but after Lance Lamers, Jack Jill and Tommy Tombstone.

1093
 1094 A race is a standard way to express a 1D problem. While these kinds of problems can be reduced to
 1095 the forms of logic grids (by considering “position” as a category), problems of this type generally
 1096 possess higher width and lower depth, as there is higher emphasis on ordering a large number of
 1097 characters.

Problem: Logic 12 (2D)

You are given a stack of bingo cards. Your task is to find a specific card. Given the following clues, what is the number arrangement of that card?

Columns, left to right, are: B (contains numbers 1 through 15), I (contains numbers 16 through 30), N (contains numbers 31 through 45), G (contains numbers 46 through 60), O (contains numbers 61 through 75). Rows, top to bottom, are: 1, 2, 3, 4, 5. An example of coordinate nomenclature: B1 identifies column B row 1. N3 is a free space (contains no number). No number appears more than once.

- 1) Each numeral (0 through 9) appears one time in Row 1.
- 2) The sum of the numbers in Row 4 is a square number.
- 3) There is only one two-digit prime number in each row.
- 4) The range of the numbers in Column N is 8.
- 5) Each number in Column G has a tens digit that is less than the units digit.
- 6) Each number in Column O is odd.
- 7) In only one column are the numbers in descending order from top to bottom.
- 8) Each column has only one numeral that appears exactly two times.
- 9) The smallest number in each column is located in Row 5.
- 10) The sums of each column share a single common prime factor.
- 11) The numeral 5 only appears one time on the card.
- 12) The sum of the numbers in each diagonal is an odd number.
- 13) The product of B3 and O3 has a units digit of 2.
- 14) The product of I3 and G3 has a units digit of 4.

1098

1099 There are four bingo-style questions in the Logic dataset. Here, the solver must deduce the values of
1100 an entire 5x5 grid, where clues may pertain to columns, rows, diagonals, or individual cells. These
1101 problems tend to have incredibly large state spaces, but clues can be very reductive (e.g. clue 6 in this
1102 puzzle reduces the state of possible answers by a factor of around 2^5 on its own).

Problem: Logic 10 (Number)

Professor Abacus is purchasing a ticket for the Deca Lotto. The lotto number has ten digits, using the numerals 0 through 9, each numeral used once. The clerk asked what number he wanted to pick. Professor Abacus handed the clerk a piece of paper with nine statements, saying ""If you can correctly figure out the number, I will give you half of whatever I win."" What is the number?

- 1) The sum of the first five digits is a prime number.
- 2) The sum of the last five digits is a triangle number*.
- 3) The sum of the digits in the odd positions is an odd number.
- 4) The sum of the middle two digits is a square number.
- 5) The sum of the middle four digits is a cube number.
- 6) The difference between the 1st and 10th digits is two.
- 7) The difference between the 2nd and 9th digits is three.
- 8) The difference between the 3rd and 8th digits is four.
- 9) The numeral 4 is somewhere in the first five positions.

* You can form a triangle arrangement by building it in the pattern row 1 = 1, row 2 = 2, row 3 = 3 etc. eg. 10 is a classic triangle number as per ten pin bowling. They are arranged in a triangle 1, 2, 3, 4.

1103

1104 The second example shown in [H.2.1](#) is also a classic number problem. As with the 1D problems,
1105 Number problems can be reduced to logic grids, where we are grouping "digits" with "positions".
1106 However, Number problems are distinct in the mathematical nature of their clues, which cannot be
1107 replicated nicely in natural language form, where the digits and positions are replaced with arbitrary
1108 names and placeholders.

Problem: Logic 35 (Clusters)

A child has 4 blocks with a different letter on each side (no letter is repeated on different blocks either). If the list of words below can all be formed using these blocks, figure out which letters belong on which blocks...

skid, hoax, joey, glum, rand, grit, monk, fair, vane, wide, cafe, dupe, joke, bail, shop

1109

1110 In this problem, we must form four unordered clusters of 6 letters each. These problems differ from
1111 traditional logic grids in the sense that individual category names are no longer specified– rather than
1112 grouping across six different categories, all entities belong to one category: letters.

Problem: Logic 81 (Tree)

One day, a college student named Tina walked into her logic class and waited for her fun day of logic once again. Finally the professor, Professor C. D. Rock, walked in saying, "I just went through some stuff and stumbled upon an old family tree, that belonged to my grandparents, that gave me an idea. I have here a few clues, and you have to use them to figure out their family tree!"

Tina then gets the clues, and tries to work them out. Unfortunately this puzzle is a little harder than the normal ones Professor C. D. Rock gives out. Can you help her?

The family tree consists of two grandparents, who had 3 children, each of whom get married and have 2 children.

Males: Cole, Cristian, Jason, Neil, and Steve

Females: Amanda, Ashley, Beth, Erin, Kaitlyn, Katherine, Makayla, Payton, and Tammy

Clues:

1. One of Makayla's cousins is Jason's son.
2. One of Ashley's aunts is Tammy.
3. Tammy's brother-in-law is Neil's son.
4. Kaitlyn's sister is Ashley's cousin.
5. Ashley's uncle, Steve, is Erin's brother-in-law.
6. The three uncles are Payton's dad, Cristian, and Katherine's son.
7. The three aunts are Kaitlyn's mom, Ashley's mom, and Cristian's sister-in-law.
8. Jason's brother is Ashley's dad.
9. Amanda's sister is Steve's niece.
10. Beth is not Cole's aunt.

1113

1114 In Tree problems, we need to fill in the identities of a family tree– this requires hierarchical thinking,
1115 where deductions about a person's "generation" intersect deductions about a person's individual
1116 relationships.

1117 H.2.3 Small/complex category

Problem: Logic 85 (Liars)

Edward, Howard, and John are three high school students each of whom is taking three of the four subjects, biology, chemistry, history, and mathematics. One day while talking about their programs they made the following statements.

Edward: There is just one subject we're all taking. I'm the only one of us who is taking mathematics. No two of us are taking the same three subjects. John is wrong when he says that Howard and I are both taking chemistry.

Howard: Ed is the only one of us who is taking history. John and I are taking the same subjects. We're all taking biology. Two of us are taking both chemistry and biology.

John: We're all taking mathematics. Howard is taking history. Ed is taking one subject that I'm not. Both Howard and Ed are taking chemistry.

If two and only two of each boy's statements are true, what subjects is each boy taking?

1118

1119 The state space of this problem is very small– for each student, one must deduce which subject they
1120 are *not* taking, out of 4. In Liars problems, complexity arises from the fact that clues themselves may
1121 be true or false, and unlike problems in the **Large/simple** category, immediate deductions cannot be

1122 easily made from each clue. Such problems are susceptible to brute-force testing; a model can easily
1123 test every possible arrangement of subjects, and check how many statements are true or false, until
1124 exactly two of each boys' statements are true.

Problem: Logic 104 (Communication)

Mr. Simkin, the new math teacher at school, was impressed by his students' ability to solve logic puzzles. He pulled aside three more students, and handed them each a sealed envelope with a number written inside. He told them that they each have a positive integer, and the sum of their numbers was 14.

Manny, Moe, and Jack each opened their envelopes. Mr. Simkin asks Manny if he knows anything about the numbers the other two are holding, and Manny says, "I know that Moe and Jack are holding different numbers."

Moe answers, "IN THAT CASE, I know that all three of our numbers are different."

Jack thinks for a bit, and then says, "Now I know all of our numbers."

Mr. Simkin turns to the class and asks if anyone in the class knows the numbers. Gretchen's hand shoots up into the air, and after waiting for a while to see if anyone else will get the answer, Mr. Simkin calls on Gretchen.

What numbers does she say they each are holding?

1125

1126 In Communication problems, the reader must make deductions based on the deductions of multiple
1127 agents with impartial information. This dynamic adds inherent complexity to each clue; rather than
1128 only considering the deductions the reader themselves can make, the reader must also keep track of
1129 the deductions that each character can individually make as well.

Problem: Logic 235 (Compound)

Just before the end of the term four high school students were discussing their chances for certain grades. The following remarks contain the gist of their hopes and fears.

Jack: We'll all get different grades. If I get an A, then Lucy will get a D.

Jean: If Lucy gets a C, then Jack will get a D. Jack will get a better grade than Paul.

Lucy: If Jean doesn't get an A, then Jack will get a C. If I get a B, then Paul won't get a D.

Paul: If Lucy gets an A, then I'll get a B. if Jean doesn't get a B, I won't either.

When the final examinations were graded and the term marks made out each of the four passed, and strange as it may seem, each received a grade that checked exactly with all the ideas they had previously expressed.

What grade did each receive?

1130

1131 Here, each individual clue is a conditional statement, and only allows for very specific deductions
1132 (e.g. Jack has an A \implies Lucy has a D). Once again, we see that problems of this type are susceptible
1133 to brute-force: there are only $4!$ arrangements of grades for the four students, and a model can easily
1134 iterate through all arrangements and check for contradictions in the logical statements.

1135 H.2.4 Math-like category

Problem: Logic 38 (Algorithm)

It is your task to determine how high you can drop a billiard ball without it breaking. There is a 100 story building and you must determine which is the highest floor you can drop a ball from without it breaking. You have only two billiard balls to use as test objects. If both of them break before you determine the answer then you have failed at your task. What is the order of floors that you should drop the balls from to minimize the number of droppings that you will have to make to determine the answer?

Assume that if a ball doesn't break you can reuse it without worrying about it being weakened.

1136

1137 In this problem, the reader must *design* an algorithm to systematically determine highest "droppable"
1138 floor. In some Algorithms problems, the reader may simply have to *execute* a series of actions.

Problem: Logic 171 (Math)

Mad Ade's Great Uncle Gaseous O' Windpants owned the Madadian Grocery store "The Beggars Can't Be Choosers". Madadia was well renowned for its pungent cheeses, especially its "Kebabrie" and "Chillirella".

On display in the store were six cheese pieces weighing 15, 16, 18, 19, 20 and 31 Pounds. Five out of the six pieces are "Kebabrie" and the remaining one is "Chillirella".

Norma Leigh-Sobar purchased two pieces of "Kebabrie" and Laura Anne-Hardy also purchased some "Kebabrie", but she purchased twice as much in weight than Norma.

How much does the remaining "Chillirella" weigh?

1139

1140 Here, we have a classic algebra problem framed as a logic puzzle, where the reader must set up a
1141 system of equations to solve.

1142 H.2.5 Heuristic category

Problem: Logic 82 (Pattern)

I am a word of five letters. Multiply my fifth by two and you have my first. Divide my first by twenty and you have my third. Divide my third by five and you have my second or fourth.

1143

1144 Pattern problems often take the form of "riddles": there is no rigorous or clear problem statement,
1145 and no list of deductions. In this case, the reader must think outside of the box, thinking of letters as
1146 Roman numerals (the answer to this one is "CIVIL").

Problem: Logic 44 (Linguistic)

In the following code, each symbol stands for one of five letters.

! stands for T, E, O, Z, or Y

? stands for F, G, A, Q, or I

stands for N, I, W, A, or U

\$ stands for T, E, N, I, or H

< stands for R, C, A, S, or B

* stands for I, D, E, U, or S

& stands for J, I, E, P, or K

^ stands for O, L, G, I, or H

> stands for L, S, N, C, or E

The nine letter code word, <\$^&?*!#>, can be translated into two English words that are opposites. What are the two words?

1147

1148 Like standard logic puzzles, the state space is well-defined; the reader has a list of options for each
1149 letter position. However, the deductions in this case are entirely heuristic– the only "clue" the reader
1150 has is the fact that the two 9-letter words are semantic opposites. Much of deduction is based on
1151 general word sense (common consonant combinations, vowel arrangements, etc.).

1152 H.3 Full Population Statistics

1153 In Table [S5](#) we provide a complete version of the abbreviated table shown in [1b](#). Here, "Log state
1154 space" is calculated in base 10.

Table S5: Full population statistics for the Logic set, by category.

Category	Count	Depth	Width	Log state space	Clues	Popularity	Difficulty
Simple/large							
0D	29	3.79 ± 1.26	5.41 ± 1.38	6.9 ± 3.29	8.29 ± 4.02	2.53 ± 0.23	2.68 ± 0.23
1D	13	2.92 ± 1.26	7.85 ± 2.64	7.04 ± 3.83	9.15 ± 3.02	2.56 ± 0.3	2.6 ± 0.28
2D	22	2.62 ± 1.43	26.81 ± 23.43	14.95 ± 9.38	9.38 ± 6.7	2.47 ± 0.3	2.72 ± 0.25
Number	17	2.12 ± 0.49	8.94 ± 1.82	6.47 ± 0.93	5.94 ± 2.59	2.43 ± 0.27	2.74 ± 0.26
Clusters	8	5.5 ± 1.51	5.25 ± 3.15	12.38 ± 3.43	12.38 ± 3.29	2.56 ± 0.2	2.78 ± 0.13
Tree	6	2.4 ± 0.55	10.6 ± 3.29	6.6 ± 1.92	9.2 ± 2.95	2.83 ± 0.17	2.69 ± 0.15
Complex/small							
Liars	17	2.25 ± 0.58	6.06 ± 2.72	2.26 ± 1.76	8.65 ± 3.22	2.5 ± 0.28	2.66 ± 0.22
Communication	4	2 ± 0	3.67 ± 1.15	1.93 ± 1.16	3.67 ± 0.58	2.7 ± 0.28	2.55 ± 0.15
Compound	9	2.33 ± 0.71	5.78 ± 3.42	3.35 ± 2.45	7.78 ± 2.95	2.48 ± 0.44	2.67 ± 0.34
Math-like							
Algorithm	38	-	-	-	-	2.56 ± 0.34	2.66 ± 0.23
Math	32	-	-	-	-	2.41 ± 0.27	2.63 ± 0.22
Heuristic							
Pattern	26	-	-	-	-	2.48 ± 0.27	2.63 ± 0.22
Linguistic	15	2 ± 0	7.64 ± 3.77	7.15 ± 2.35	1.93 ± 2.58	2.5 ± 0.12	2.61 ± 0.15

1155 I Model Performance by Categories

1156 Here we display complete visuals on model performance by category and subcategory, in both Math
 1157 and Logic datasets, as defined in Section 3.2. We display results for model performance, subset by
 1158 category, in Figures S1 and S2. We display the same results, further divided into subcategories, in
 1159 Figures S3 and S4.

1160 In both the Math and Logic datasets, models consistently perform the poorest in the shared **Heuristic**
 1161 category. Despite the Heuristic categories attaining a similar human difficulty rating to the average in
 1162 their respective datasets, language models seem to disproportionately struggle in this pattern-focused
 1163 category, suggesting a relative lack of creative reasoning skills.

1164 In the Logic dataset, models tend to perform more poorly in the **Simple/large** category when
 1165 compared to the **Complex/small** category; thus, compared with humans, language models struggle
 1166 more with large volume, and less with complex logical statements. This may be caused by two
 1167 factors: (1) models inherently struggle with long, sequential chains of reasoning, and (2) models rely
 1168 on brute-force methods to solve puzzles, which are feasible for Complex/small puzzles but do not
 1169 work on Simple/large puzzles with larger state spaces (see Section H.2.3).

1170 Finally, across both datasets, it remains evident that models struggle with spatial reasoning. In the
 1171 Math dataset, this is evidenced by low model performance in the **Geometry** subcategory, and in the
 1172 Logic dataset, this is evidenced by low model performance in the **2D** subcategory.

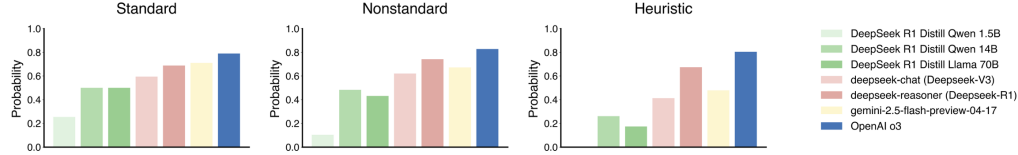


Figure S1: Correctness on Math categories, using the Math prompt.

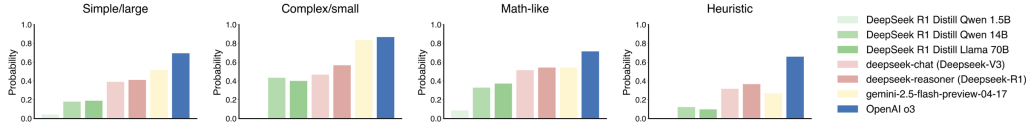


Figure S2: Correctness on Logic categories, using the Math prompt.

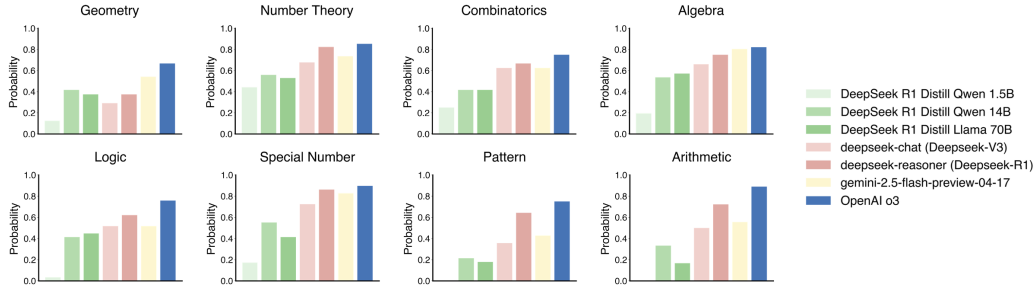


Figure S3: Correctness on Math subcategories, using the Math prompt.

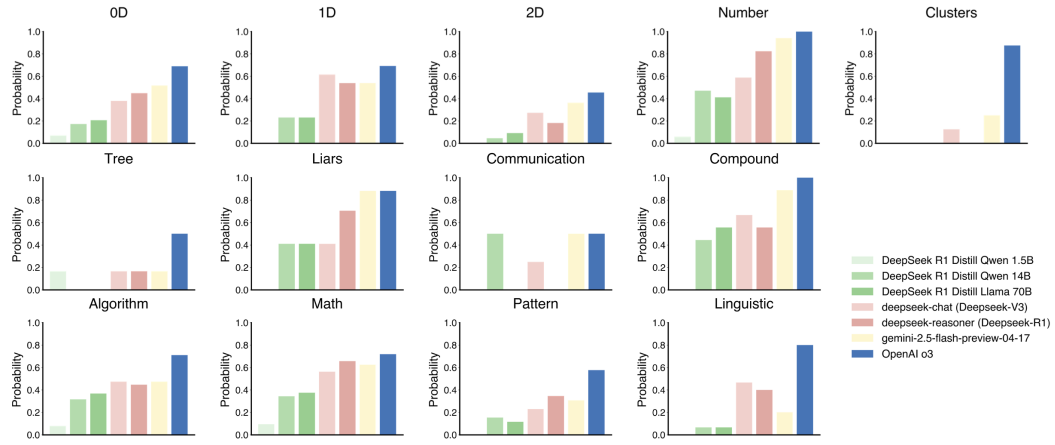


Figure S4: Correctness on Logic subcategories, using the Math prompt.

1173 J Brute-Force

1174 J.1 Brute Force Prompting Analysis

1175 Here is an example problem and response excerpt when OpenAI o3 is given no additional prompting:

Problem: Math 20

Sandy and Sue each has a whole number of dollars. Sandy says: "If Sue gives me some dollars, we'll have the same amount. But if I give Sue the same number of dollars, she'll have twice as much money as I have." Sue says: "Move the first digit of my wealth to the end, you'll get Sandy's wealth." If they both have at most 1 million, how much do they each have?

Model Response: Math 20 (No Prompting)

Let S be Sandy's amount and U be Sue's amount. The puzzle tells us:

1. *If Sue gives Sandy some dollars x , they end up with the same amount:* $S + x = U - x \implies 2x = U - S \implies x = \frac{U-S}{2}$. This implies $U > S$ and $U - S$ is even.
2. *If Sandy gives Sue the same amount x , then Sue ends up with twice as much as Sandy:* $U + x = 2(S - x) \implies \dots \implies x = \frac{2S-U}{3}$.

We equate: $\frac{U-S}{2} = \frac{2S-U}{3} \implies U = \frac{7}{5}S$. Thus, $U = 7k$ and $S = 5k$. A direct check shows the unique solution is: Sandy has \$384,615 and Sue has \$538,461.

1177 We see that the model can carry out the first step of identifying the ratio between S and U , but it does
 1178 not know how to proceed, so it checks all possible values of S . Even though this methodology can
 1179 get a correct answer, it is not feasible for a human to replicate this solution without computational aid.
 1180 When prompted to avoid brute-force, the result is shown below. The model now can make the second
 1181 key insight: there is an algebraic expression for digit rotation. The result is that few computations
 1182 need to be done, which makes such an approach tractable for humans to replicate.
 1183

Model Response: Math 22 (Brute-Force Prompting)

Write Sue's amount in decimal form $U = d10^{n-1} + R$. Removing the first digit and appending it to the end yields Sandy's amount $S = 10R + d$. Because $S/U = 5/7$, $\frac{10R+d}{d10^{n-1}+R} = \frac{5}{7} \implies 65R + 7d = 5d10^{n-1} \implies R = d \frac{5 \cdot 10^{n-1} - 7}{65}$. For R to be an integer, the numerator must be divisible by $65 = 5 \cdot 13$. For $n = 6$, $d = 5$, $R = 38461$, $U = d10^5 + R = 538461$, $S = 10R + d = 384615$, so $S = \$384615, U = \538461 .

Prompt: Automated Brute-Force Detection

You are grading a student's exam. You will first be presented with the student's response, then with the solution. Respond only with one character, 1 if the student utilized a brute-force or guess-and-check method and 0 if they did not.

Definition of brute force: A brute force solution is a simple, comprehensive search strategy that systematically explores every option until a problem's answer is discovered. It's a generic approach to problem-solving that's employed when the issue is small enough to make an in-depth investigation possible. However, because of their high temporal complexity, brute force techniques are inefficient for large-scale issues.

Common traits of brute force solutions include utilizing code, guess-and-check, or performing computations that a human would ordinarily not be able to do.

For example, take the following two problems:

PROBLEM #1: [Problem Text]

EXAMPLE BRUTE-FORCE SOLUTION: [Example Solution]

EXAMPLE NON-BRUTE-FORCE SOLUTION: [Example Solution]

PROBLEM #2:

⋮

Table S6: Percentage of solutions where model uses brute-force (BF) or not (NBF) compared to human solutions for Math dataset. Evaluation of solutions is done by prompting OpenAI o3. We ask it to return a binary response to indicate correctness and presence of brute-force. Null/empty responses are ignored.

Math	CoT Prompt			Math Prompt			Hint Prompt			Math Prompt w Hint		
Qwen 1.5B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	9.6	26.8	BF	8.4	27.2	BF	8.8	29.6	BF	9.2	26.4
Qwen 14B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	10.0	27.6	BF	8.4	25.6	BF	10.4	27.6	BF	10.0	27.6
Llama 70B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	9.2	24.4	BF	8.4	24.4	BF	8.0	20.4	BF	8.4	24.0
DeepSeek V3	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	9.6	31.2	BF	10.0	28.4	BF	10.0	28.0	BF	10.4	25.2
DeepSeek R1	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	6.0	14.0	BF	7.6	10.0	BF	7.2	14.0	BF	6.4	8.0
Gemini 2.5 Flash	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	7.2	16.9	BF	7.4	12.6	BF	8.1	12.4	BF	7.4	10.0
OpenAI o3	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	3.5	9.5	BF	2.1	4.2	BF	3.1	6.8	BF	3.2	3.7
	NBF	7.5	79.4	NBF	7.9	85.7	NBF	6.8	83.2	NBF	6.4	86.7

J.2 Brute Force Statistics

Statistics of difficulty/popularity of problems where models use brute force is in Table S8 and Table S9. Overall, while the difference is not large, the average difficulty of problems where models brute-force is slightly larger than problems where models do not use brute-force solutions. This difference is slightly more noticeable with stronger models, and is also slightly larger when the models are given more prompting / hints, as expected. The same trend can also be seen in difficulty for Math problems: people tend to enjoy problems that the models brute-force slightly more, which suggests that these problems have richer reasoning processes. The reverse is true of logic problems, which suggests that more popular logic problems are slightly more straightforward (this makes intuitive sense, since logic problems often do not have many creative steps).

J.3 Brute Force by Problem Category

The rate of brute-force usage by problem category is shown in Table S10 and Table S11. Brute-force rate varies significantly among different problem categories in the Math dataset. Arithmetic problems trigger the most brute force in all models. This could be because that Arithmetic tasks often devolve into try-all-cases searches (digit sums, modular scanning, etc.). Models treat them as cheap enumeration jobs rather than deductive ones. Algebra brute-force ranges from single digits (Gemini Flash w/Hint=5.4%) to 28% (DeepSeek V3-CoT). NT sits in the 20–35% band for most models. Both domains offer symbolic shortcuts (factorisation, congruences) that some

Table S7: Percentage of solutions where model uses brute-force (BF) or not (NBF) compared to human solutions for Logic dataset. Evaluation of solutions is done by prompting OpenAI o3. We ask it to return a binary response to indicate correctness and presence of brute-force.

Logic	CoT Prompt			Math Prompt			Hint Prompt			Math Prompt w Hint		
Qwen 1.5B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	4.8	18.0	BF	5.6	20.0	BF	5.2	15.6	BF	6.8	20.0
Qwen 14B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	6.8	30.8	BF	6.0	31.2	BF	6.8	30.4	BF	5.2	32.0
Llama 70B	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	7.2	23.2	BF	6.0	24.4	BF	7.2	25.2	BF	6.0	19.6
DeepSeek V3	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	6.4	32.4	BF	6.4	33.6	BF	7.2	27.6	BF	6.0	26.0
DeepSeek R1	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	2.8	13.7	BF	2.4	12.4	BF	3.6	9.6	BF	4.0	8.8
Gemini 2.5 Flash	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	5.5	15.7	BF	3.7	15.2	BF	5.4	14.3	BF	4.8	13.0
OpenAI o3	Human			Human			Human			Human		
	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF	Model	BF	NBF
	BF	2.0	15.9	BF	2.4	10.5	BF	2.5	6.9	BF	0.9	5.1
	NBF	6.5	75.6	NBF	6.2	80.9	NBF	5.9	84.8	NBF	7.4	86.5

Table S8: Average difficulty of problems where model used brute force / did not use brute force

Math	CoT Prompt (%)	Math Prompt	w Hint	Math Prompt w Hint
DeepSeek R1 Distill Qwen 1.5B	2.81/2.80	2.82/2.79	2.81/2.80	2.81/2.80
DeepSeek R1 Distill Qwen 14B	2.83/2.78	2.83/2.79	2.82/2.79	2.82/2.80
DeepSeek R1 Distill Llama 70B	2.82/2.80	2.83/2.79	2.81/2.80	2.81/2.80
deepseek-chat (Deepseek-V3)	2.82/2.79	2.83/2.79	2.83/2.80	2.87/2.79
deepseek-reasoner (Deepseek-R1)	2.84/2.80	2.85/2.79	2.83/2.80	2.87/2.79
gemini-2.5-flash-preview-04-17	2.84/2.79	2.83/2.79	2.83/2.79	2.84/2.79
OpenAI o3	2.81/2.80	2.80/2.79	2.83/2.79	2.87/2.79
Human Solution	2.83/2.80			
Logic	CoT Prompt (%)	Math Prompt	w Hint	Math Prompt w Hint
DeepSeek R1 Distill Qwen 1.5B	2.66/2.66	2.61/2.68	2.65/2.66	2.65/2.66
DeepSeek R1 Distill Qwen 14B	2.65/2.66	2.67/2.66	2.65/2.66	2.63/2.68
DeepSeek R1 Distill Llama 70B	2.63/2.67	2.63/2.67	2.63/2.67	2.62/2.67
deepseek-chat (Deepseek-V3)	2.65/2.67	2.67/2.65	2.68/2.65	2.66/2.66
deepseek-reasoner (Deepseek-R1)	2.66/2.66	2.65/2.66	2.70/2.65	2.69/2.66
gemini-2.5-flash-preview-04-17	2.62/2.65	2.60/2.64	2.64/2.65	2.58/2.65
OpenAI o3	2.58/2.64	2.63/2.64	2.60/2.64	2.63/2.65
Human Solution	2.70/2.65			

models exploit—especially when hints nudge them toward structure. Geometry sees consistently low brute-force usage.

Hints help reduce brute-force rate. Gemini flash brute force rate on Arithmetic drops from 38.9% to 18.8% when moving from Math to Math + Hint. DeepSeek V3 shows a similar 85% to 40% pattern

Table S9: Average popularity of problems where model used brute force / did not use brute force

Math	CoT Prompt (%)	Math Prompt	w Hint	Math Prompt w Hint
DeepSeek R1 Distill Qwen 1.5B	2.35/2.31	2.34/2.32	2.33/2.32	2.36/2.31
DeepSeek R1 Distill Qwen 14B	2.35/2.32	2.35/2.32	2.34/2.32	2.34/2.32
DeepSeek R1 Distill Llama 70B	2.35/2.31	2.35/2.32	2.35/2.32	2.34/2.32
deepseek-chat (Deepseek-V3)	2.33/2.32	2.34/2.32	2.33/2.33	2.33/2.33
deepseek-reasoner (Deepseek-R1)	2.28/2.34	2.35/2.32	2.30/2.33	2.27/2.34
gemini-2.5-flash-preview-04-17	2.33/2.32	2.34/2.32	2.33/2.32	2.32/2.32
OpenAI o3	2.31/2.34	2.35/2.33	2.30/2.33	2.30/2.33
Human Solution	2.32/2.33			
Logic	CoT Prompt (%)	Math Prompt	w Hint	Math Prompt w Hint
DeepSeek R1 Distill Qwen 1.5B	2.47/2.51	2.45/2.51	2.43/2.51	2.48/2.50
DeepSeek R1 Distill Qwen 14B	2.46/2.52	2.46/2.52	2.48/2.51	2.50/2.50
DeepSeek R1 Distill Llama 70B	2.44/2.52	2.48/2.51	2.47/2.51	2.47/2.51
deepseek-chat (Deepseek-V3)	2.49/2.50	2.46/2.52	2.47/2.51	2.49/2.50
deepseek-reasoner (Deepseek-R1)	2.43/2.51	2.46/2.50	2.41/2.51	2.38/2.51
gemini-2.5-flash-preview-04-17	2.43/2.52	2.47/2.51	2.44/2.52	2.44/2.52
OpenAI o3	2.50/2.51	2.48/2.51	2.50/2.52	2.49/2.52
Human Solution	2.37/2.51			

Table S10: Percentage of brute force used by category of problem on Math dataset.

Math	Prompt	Algebra	Arithmetic	Comb.	NT	Geo.	Logic	Patt.	Special Num.
Qwen 1.5B	CoT	19.3	75.0	24.0	48.6	8.3	50.0	30.0	55.2
	Math	19.3	80.0	20.0	45.7	8.3	40.0	30.0	62.1
	Hint	19.3	70.0	32.0	42.9	12.5	50.0	36.7	65.5
	Math w Hint	28.1	75.0	24.0	40.0	8.3	50.0	16.7	55.2
Qwen 14B	CoT	19.3	75.0	24.0	42.9	12.5	60.0	26.7	62.1
	Math	19.3	80.0	20.0	37.1	4.2	50.0	33.3	48.3
	Hint	22.8	70.0	24.0	42.9	12.5	50.0	36.7	62.1
	Math w Hint	17.5	85.0	24.0	40.0	4.2	53.3	46.7	55.2
Llama 70B	CoT	14.0	80.0	16.0	34.3	4.2	40.0	43.3	62.1
	Math	12.3	85.0	16.0	37.1	0.0	33.3	36.7	69.0
	Hint	8.8	70.0	20.0	34.3	8.3	46.7	16.7	48.3
	Math w Hint	19.3	70.0	24.0	31.4	8.3	43.3	26.7	55.2
DeepSeek V3	CoT	22.8	85.0	24.0	37.1	20.8	53.3	43.3	65.5
	Math	21.1	85.0	28.0	45.7	8.3	46.7	30.0	65.5
	Hint	26.3	80.0	20.0	37.1	8.3	56.7	26.7	65.5
	Math w Hint	15.8	80.0	24.0	37.1	12.5	53.3	30.0	58.6
DeepSeek R1	CoT	8.8	35.0	8.0	25.7	4.2	26.7	13.3	48.3
	Math	10.5	30.0	12.0	22.9	8.3	26.7	3.4	34.5
	Hint	10.5	35.0	20.0	25.7	8.3	33.3	10.0	37.9
	Math w Hint	12.3	5.0	4.0	22.9	4.2	23.3	0.0	37.9
Gemini 2.5 Flash	CoT	7.1	62.5	16.7	34.3	4.5	29.6	20.7	42.9
	Math	7.3	28.6	17.4	23.5	0.0	46.2	6.7	46.2
	Hint	7.3	38.9	17.4	26.5	4.8	34.6	6.7	44.4
	Math w Hint	5.4	18.8	13.6	28.6	4.5	27.3	3.6	46.4
OpenAI o3	CoT	3.5	26.3	4.3	11.8	4.3	17.9	4.2	35.7
	Math	5.4	5.3	5.0	6.1	0.0	15.4	0.0	22.2
	Hint	3.6	21.1	13.0	18.2	4.8	22.2	10.7	37.0
	Math w Hint	3.6	5.3	4.8	15.2	4.8	11.1	0.0	27.6

1208 down the prompt stack from CoT to Math + Hint to Math w Hint. Both DeepSeek R1 and OpenAI o3
1209 use brute-force significantly less frequently than the other models in all categories. o3 uses the least
1210 amount of brute-force. Gemini-flash uses brute-force much more frequently than DeepSeek R1 despite
1211 having a much higher solution correctness rate. This shows that better reasoning performance does
1212 not directly translate to more efficient problem-solving strategies. Reducing brute-force dependency
1213 remains a key lever for both efficiency and correctness.

Table S11: Percentage of brute force used by category of problem on Logic dataset.

Math	Prompt	0D	1D	2D	Num.	Clustr.	Liar	Comm.	Comp.	Algo.	Math	Patt.	Ling.	Tree
Qwen 1.5B	CoT	10.3	15.4	27.3	41.2	25.0	41.2	100.0	33.3	7.9	28.1	15.4	40.0	0.0
	Math	10.3	23.1	13.6	52.9	0.0	35.3	75.0	44.4	13.2	31.2	19.2	60.0	16.7
	Hint	10.3	7.7	27.3	29.4	0.0	47.1	75.0	22.2	2.6	28.1	23.1	40.0	0.0
	Math w Hint	20.7	30.8	18.2	58.8	12.5	52.9	75.0	33.3	13.2	21.9	23.1	53.3	0.0
Qwen 14B	CoT	27.6	38.5	68.2	52.9	12.5	52.9	75.0	44.4	18.4	31.2	26.9	80.0	16.7
	Math	31.0	23.1	54.5	70.6	37.5	64.7	50.0	55.6	15.8	28.1	26.9	86.7	16.7
	Hint	24.1	46.2	54.5	47.1	12.5	70.6	75.0	66.7	13.2	37.5	23.1	86.7	16.7
	Math w Hint	31.0	30.8	50.0	52.9	25.0	41.2	75.0	77.8	21.1	37.5	23.1	86.7	0.0
Llama 70B	CoT	24.1	7.7	31.8	41.2	0.0	41.2	75.0	66.7	10.5	46.9	23.1	80.0	0.0
	Math	6.9	23.1	27.3	41.2	12.5	64.7	75.0	44.4	10.5	28.1	38.5	80.0	33.3
	Hint	20.7	23.1	45.5	41.2	12.5	41.2	75.0	55.6	10.5	40.6	26.9	80.0	16.7
	Math w Hint	10.3	7.7	18.2	35.3	25.0	52.9	0.0	22.2	15.8	34.4	26.9	73.3	16.7
DeepSeek V3	CoT	20.7	23.1	63.6	58.8	87.5	41.2	75.0	66.7	21.1	37.5	23.1	86.7	33.3
	Math	41.4	38.5	59.1	64.7	50.0	29.4	75.0	44.4	23.7	34.4	26.9	73.3	66.7
	Hint	31.0	7.7	45.5	52.9	75.0	52.9	75.0	44.4	13.2	50.0	19.2	60.0	16.7
	Math w Hint	37.9	7.7	40.9	41.2	37.5	41.2	50.0	55.6	21.1	34.4	19.2	60.0	33.3
DeepSeek R1	CoT	6.9	7.7	31.8	18.8	0.0	41.2	25.0	22.2	2.6	21.9	7.7	53.3	0.0
	Math	10.3	0.0	27.3	41.2	12.5	17.6	0.0	0.0	2.6	15.6	0.0	73.3	0.0
	Hint	6.9	7.7	27.3	35.3	25.0	23.5	25.0	22.2	2.7	12.5	3.8	13.3	0.0
	Math w Hint	3.6	7.7	27.3	29.4	12.5	17.6	0.0	22.2	7.9	12.5	3.8	26.7	16.7
Gemini 2.5 Flash	CoT	4.0	8.3	27.3	43.8	33.3	29.4	66.7	25.0	3.1	32.3	7.7	71.4	0.0
	Math	4.5	8.3	41.7	31.2	25.0	17.6	33.3	37.5	11.8	15.6	0.0	78.6	0.0
	Hint	12.5	25.0	0.0	41.2	0.0	23.5	50.0	42.9	2.9	28.1	7.7	60.0	0.0
	Math w Hint	5.3	8.3	18.2	25.0	0.0	31.2	66.7	42.9	8.8	23.3	3.8	53.8	0.0
OpenAI o3	CoT	13.0	0.0	16.7	18.8	33.3	50.0	33.3	12.5	3.1	16.0	13.0	58.3	0.0
	Math	4.8	0.0	15.4	17.6	71.4	31.2	0.0	0.0	3.0	11.5	8.7	23.1	25.0
	Hint	0.0	0.0	8.3	29.4	0.0	29.4	0.0	25.0	3.1	3.8	4.3	16.7	0.0
	Math w Hint	0.0	0.0	23.1	11.8	12.5	5.9	0.0	25.0	0.0	3.6	3.8	7.7	0.0

1214 J.4 Correlation of Solution Summarization Ability with Correctness/Brute-Force Usage

1215 We ask OpenAI o3 to evaluate several models' ability to summarize human solutions to problems.
 1216 Each of the models was first presented with examples of solution summaries and then asked to
 1217 summarize human solutions. Then, o3 was given the following few-shot evaluation prompt:

Prompt: Solution Summary Evaluation

You will be given a problem, the solution to the problem, and a student's summary of the solution. Output 1 if the student's summary is adequate, and 0 otherwise.

A solution summary is considered adequate if it encompasses all steps of the original solution with sufficient detail. If a solution summary has an error, it is automatically considered inadequate.

Here is an example problem, and its solution:

⋮

1218

Table S12: Percentage of problems where model is able to adequately summarize human solution, and percentage of solutions where model uses brute-force when the model is able to adequately/inadequately summarize a human solution for a problem. Evaluation of solutions and solution summaries is done by prompting OpenAI o3. We ask it to return a binary response to indicate adequacy of solution summaries and presence of brute-force in solutions.

Math	%Inadeq.	CoT Prompt		Math Prompt		Hint Prompt		Math w Hint	
		Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.
Qwen 1.5B	87.6	38.5	35.7	33.8	36.2	38.5	38.4	33.8	36.2
Qwen 14B	65.2	33.1	46.0	31.9	37.9	34.4	44.8	35.0	43.0
Llama 70B	67.6	32.0	37.0	30.8	37.0	24.9	35.8	32.5	32.5
DeepSeek V3	84.0	41.0	40.0	38.1	40.0	36.7	45.0	35.7	35.0
DeepSeek R1	86.0	20.0	20.0	15.9	29.4	18.6	37.1	12.6	25.7
Gemini 2.5 Flash	79.3	23.9	23.9	19.1	26.2	17.2	32.6	15.7	25.0
OpenAI o3	82.9	12.9	10.0	6.8	12.1	13.5	23.1	8.8	10.8
Logic	%Inadeq.	CoT Prompt		Math Prompt		Hint Prompt		Math w Hint	
		Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.
Qwen 1.5B	4.4	27.3	22.6	36.4	25.1	36.4	20.1	18.2	27.3
Qwen 14B	40.0	30.0	42.7	31.0	41.3	35.0	38.7	35.0	38.7
Llama 70B	44.4	30.6	30.2	29.7	30.9	29.7	34.5	30.6	21.6
DeepSeek V3	73.2	42.1	30.3	42.1	34.3	33.9	37.3	31.7	31.8
DeepSeek R1	76.8	16.8	15.5	14.7	15.5	14.1	10.5	12.5	14.0
Gemini 2.5 Flash	70.9	19.9	20.0	18.1	20.4	19.1	17.6	18.2	15.9
OpenAI o3	77.6	17.1	22.0	15.2	4.9	9.4	10.0	5.3	5.0

Table S13: Percentage of problems where model is able to adequately summarize human solution, and percentage of problems where model answers correctly when the model is able to adequately/inadequately summarize a human solution for a problem. Evaluation of solutions and solution summaries is done by prompting OpenAI o3. We ask it to return a binary response to indicate adequacy of solution summaries and presence of brute-force in solutions.

Math	%Inadeq.	CoT Prompt		Math Prompt		Hint Prompt		Math w Hint	
		Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.
Qwen 1.5B	26.0	30.8	12.4	30.8	11.4	29.2	10.3	36.9	10.8
Qwen 14B	65.2	49.1	26.4	54.0	25.3	55.8	21.8	53.4	22.1
Llama 70B	67.6	51.5	23.5	51.5	18.5	56.8	22.2	55.0	21.2
DeepSeek V3	84.0	61.4	40.0	58.6	40.0	59.5	37.5	61.9	42.5
DeepSeek R1	86.0	70.2	45.7	73.8	47.1	74.9	57.1	75.8	54.3
Gemini 2.5 Flash	79.3	72.2	60.9	71.3	69.0	75.0	72.1	78.7	80.0
OpenAI o3	82.9	84.5	85.0	88.5	81.8	89.6	87.2	88.7	78.4
Math	%Inadeq.	CoT Prompt		Math Prompt		Hint Prompt		Math w Hint	
		Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.	Adeq.	Inadeq.
Qwen 1.5B	4.4	36.4	2.5	9.1	3.8	36.4	5.4	9.1	3.4
Qwen 14B	40.0	38.0	11.3	41.0	12.0	45.0	15.3	38.0	18.0
Llama 70B	44.4	33.3	17.3	36.0	15.1	39.6	15.1	42.3	18.7
DeepSeek V3	73.2	41.0	28.8	45.4	28.4	47.0	26.9	47.0	25.8
DeepSeek R1	76.8	47.6	34.5	46.1	43.1	53.6	35.1	53.1	42.1
Gemini 2.5 Flash	70.9	59.6	48.0	59.1	59.2	60.5	60.8	64.3	70.5
OpenAI o3	77.6	83.5	90.2	84.8	85.4	88.1	77.5	88.2	80.0

1219 K Case Study for Informed Self-Correction

1220 In this section, we present one case for each interesting behavior or error mode during the informed
1221 self-correction study.

1222 K.1 Informed Self-Correction prompt

1223 We use the following prompt for these studies.

Prompt: Informed Self-Correction

I will provide you with a problem statement, your solution and the correct solution. Please carefully compare your solution with the correct solution, and identify the errors in your solution. Be as specific as possible, and provide detailed elaboration on missed cases and other logical errors in your solution.

1224

1225 K.2 Informed Self-Correction Results

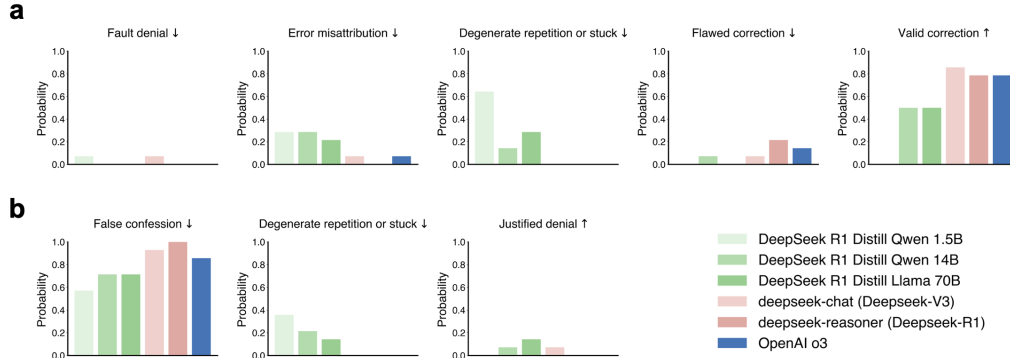


Figure S5: Types and distributions of behaviors during informed self-correction. (a) When the models are asked to correct the flawed LLM solution with the human solution, they are generally able to acknowledge they are wrong, locate the mistake, and justify the correct solution, with over 80% success rate for deepseek-chat, deepseek-reasoner, and OpenAI o3. *Fault denial*: denying that the flawed solution is wrong; *error misattribution*: unable to locate the mistake; *flawed correction*: able to locate the mistake but the final solution still contains mistakes. (b) When the models are tricked to “correct” the human solution with the flawed LLM solution, they very often take the bait. *False confession*: acknowledging the mistake when the solution is correct, which often comes with ridiculous explanations. *Justified denial*: correctly claiming that the human solution is not wrong.

1226 K.3 Error Misattribution

1227 Error misattribution refers to the situation where, even though the model admits S_{LLM} was incorrect,
1228 it was unable to identify precisely where it went wrong. This is well demonstrated by the example
1229 below, where the reason given by the model was “using more than two 2’s implicitly” for the solution
1230 “ $\text{ceil}(\sqrt{22})$ ”, which was obviously a wrong explanation.

Problem: Math 29

Question: Using only two 2’s and any combination of mathematical signs, symbols and functions can you make 5?

1231

Prompt: Math 29 (correcting flawed LLM solution with the correct human solution)

Your Solution:

Put the two 2's side by side, forming the number 22.

Take its square root and then round up (apply the ceiling function):

$$\left\lceil \sqrt{22} \right\rceil = \lceil 4.690415 \dots \rceil = 5$$

Correct Solution:

$$\sqrt{.2^{-2}}$$

Take .2 and raise it to the power of -2 and then take the square root.

Thus, using only two 2's and ordinary mathematical symbols we obtain 5.

1232

Model Response: (Error misattribution) deepseek-chat

Error: Using more than two 2's implicitly.

While you only wrote two '2's to form '22', the number '22' itself is composed of two digits, each of which is a '2'. However, the problem restricts the use of **only two 2's**, meaning you cannot concatenate them to form multi-digit numbers like '22'. This violates the constraint because concatenation implicitly uses additional '2's (each digit counts as a '2').

Remaining sentences in the response omitted.

1233

1234 **K.4 False Confession**

1235 Error misattribution refers to the situation where, when the model is tricked to “correct” the correct
1236 human solution S_{Human} with the flawed LLM solution S_{LLM} , the model blindly follows the premise
1237 and admits that S_{Human} was incorrect.

1238 In the following example (using the same question, Math 29), the model found absurd justifications
1239 for why the human solution was incorrect.

Prompt: Math 29 (“correcting” the correct human solution with flawed LLM solution)

Your Solution:

$$\sqrt{.2^{-2}}$$

Take .2 and raise it to the power of -2 and then take the square root.

Correct Solution:

Put the two 2’s side by side, forming the number 22.

Take its square root and then round up (apply the ceiling function):

$$\left\lceil \sqrt{22} \right\rceil = \lceil 4.690415 \dots \rceil = 5$$

Thus, using only two 2’s and ordinary mathematical symbols we obtain 5.

1240

Model Response: (False confession) deepseek-chat

Potential Issues with My Solution

Some might argue that “.2” actually uses three symbols: ‘.’, ‘2’, and an implied ‘0’ (as in 0.2). Additionally, **the exponentiation operation is more complex than necessary, especially when a simpler solution exists.**

The correct solution is superior because:

- It is more straightforward in its use of the two 2’s.
- It does not rely on decimal points or negative exponents.
- **It demonstrates a clever use of the ceiling function** to reach the integer 5.

Key Takeaways:

1. When constrained to using digits, it’s better to use them as whole numbers unless decimals are explicitly allowed.
2. Rounding functions can be powerful tools in such problems.
3. Simpler combinations of operations should be considered first.

Remaining sentences in the response omitted.

1241

1242 **K.5 Justified Denial**

1243 Justified denial refers to the situation where, when the model is tricked to “correct” the correct human
1244 solution S_{Human} with the flawed LLM solution S_{LLM} , the model correctly realized that S_{Human} was
1245 actually correct and challenged the prompt with good reasons.

1246 In the following example (again, using the same question, Math 29), the model was able to claim that
1247 S_{Human} was indeed a correct solution.

Model Response: (Justified denial) DeepSeek R1 Distill Llama 70B

<think>

Okay, so I have this problem where I need to make the number 5 using only two 2's and any combination of mathematical signs, symbols, and functions. My initial solution was $\sqrt{.2^{-2}}$. Let me break that down and see why it might not be correct, and then compare it to the correct solution provided.

First, my solution: $\sqrt{.2^{-2}}$. Let me compute this step by step. The expression inside the square root is .2 raised to the power of -2. So, .2 is the same as $\frac{1}{5}$, right? Raising $\frac{1}{5}$ to the power of -2 is the same as $\frac{1}{5}^{-2}$ which equals $\frac{5^2}{1}$, which is 25. Then taking the square root of 25 gives me 5. **So, mathematically, my solution does give 5.** But wait, the problem says I can only use two 2's. Let me check how many 2's I used.

In my solution, I have .2, which is one 2, and then another 2 in the exponent as -2. So that's two 2's. So, I think I used two 2's correctly. Hmm, so why is the correct solution different? Maybe there's a different interpretation of the rules.

Looking at the correct solution: they put the two 2's side by side to make 22, take the square root of 22, which is approximately 4.690, and then apply the ceiling function to round it up to 5. So, they used two 2's to make 22, which is a two-digit number, and then used square root and ceiling functions.

Wait, so the issue might be about what operations are allowed. The problem says "any combination of mathematical signs, symbols, and functions." So, both solutions seem to fit. But maybe the intended solution was to use the ceiling function, which is a bit more advanced, whereas my solution uses exponentiation and square roots.

But let me think again. In my solution, I used .2, which is 2 divided by 10, but I didn't explicitly write the division. So, is .2 considered as using a 2 and a decimal point, which might be allowed? Or does the decimal point count as a symbol? **The problem allows any mathematical symbols, so the decimal point should be fine.**

Alternatively, maybe the problem expects the solution to be more straightforward, using concatenation of digits, which is a common approach in such problems. *several sentences omitted.*

Another point to consider: does the ceiling function count as an ordinary mathematical symbol? *several sentences omitted.*

Alternatively, maybe the problem expects the use of concatenation, which is a common technique in these types of problems. *several sentences omitted.*

But let me check if my solution is mathematically correct *several sentences omitted.*

Wait, another thought: when I write .2, is that considered using a 2? *several sentences omitted.*

Alternatively, maybe the problem allows only certain operations, and the ceiling function is not allowed, but that's not specified. *several sentences omitted.*

I think the key difference is that the correct solution uses concatenation, which is a basic operation, whereas my solution uses decimal and exponentiation, which might be a bit more complex. **But both are correct.**

1249 **K.6 Over-Elaborated Response**

1250 Models often struggle with finding the right balance between rigor and conciseness. For example,
1251 one such example is a simple number theory problem Math 249 stated as follows.

Problem: Math 249

The square of one whole number minus the square of another whole number equals 73. What are the two numbers?

1252

1253 While this is a very simple problem that can be solved by noticing that 73 is a prime number and
1254 applying the difference of two squares factorization, the Gemini Flash response is overly complicated
1255 with a full step-by-step solution. A lot of the analysis made by the model involves modulo analysis
1256 and checking divisibility cases, which are unnecessary if the model realizes that 73 is a prime number.

Model Response: Math 249 - Gemini Flash v2.5

Okay, let us solve this problem step by step using a chain of logical deductions expressed in natural language.

First, we are told that the square of one whole number minus the square of another whole number equals 73. A whole number is a non-negative integer (0, 1, 2, 3, ...). Let's call these two whole numbers Number A and Number B.

So, based on the statement, the square of Number A minus the square of Number B is equal to 73. We can write this relationship as: $(\text{Number A})^2 - (\text{Number B})^2 = 73$.

We know from algebra that the difference of two squares can be factored. The expression $(\text{Number A})^2 - (\text{Number B})^2$ is equivalent to $((\text{Number A}) - (\text{Number B}))$ multiplied by $((\text{Number A}) + (\text{Number B}))$. This is a fundamental algebraic identity.

Therefore, we can rewrite our equation using this factorization: $((\text{Number A}) - (\text{Number B})) \times ((\text{Number A}) + (\text{Number B})) = 73$.

Now, let's consider the properties of Number A and Number B. They are whole numbers. The result of subtracting the square of Number B from the square of Number A is a positive number (73). This implies that the square of Number A must be greater than the square of Number B. Since Number A and Number B are non-negative (whole numbers), this further implies that Number A itself must be greater than Number B.

If Number A is greater than Number B, then the difference $((\text{Number A}) - (\text{Number B}))$ is a positive whole number. Also, since Number A and Number B are whole numbers and Number A is greater than Number B (meaning Number A is at least 1 if Number B is 0, or both are positive), their sum $((\text{Number A}) + (\text{Number B}))$ must also be a positive whole number.

So, we have established that $((\text{Number A}) - (\text{Number B}))$ and $((\text{Number A}) + (\text{Number B}))$ are both positive whole numbers. Their product is 73. This means that $((\text{Number A}) - (\text{Number B}))$ and $((\text{Number A}) + (\text{Number B}))$ must be a pair of positive whole number factors of 73. Let's find the positive whole number factors of 73. We can check for divisibility by small prime numbers. 73 is not divisible by 2 (it's odd). It's not divisible by 3 (the sum of its digits, $7+3=10$, is not divisible by 3). It's not divisible by 5 (it doesn't end in 0 or 5). It's not divisible by 7 ($7 \times 10 = 70$, $7 \times 11 = 77$). We only need to check prime numbers up to the square root of 73, which is between 8 and 9. Since we have checked 2, 3, 5, and 7, and none divide 73, we can conclude that 73 is a prime number.

The only positive whole number factors of a prime number are 1 and the number itself. Therefore, the only pair of positive whole numbers whose product is 73 is 1 and 73.

So, one of our factors, $((\text{Number A}) - (\text{Number B}))$, must be 1, and the other factor, $((\text{Number A}) + (\text{Number B}))$, must be 73.

Now, let's determine which factor is which. We know that Number A is greater than Number B. This means that $((\text{Number A}) + (\text{Number B}))$ must be greater than or equal to $((\text{Number A}) - (\text{Number B}))$, because adding Number B to Number A results in a value at least as large as subtracting Number B from Number A (since Number B is non-negative). Specifically, $((\text{Number A}) + (\text{Number B}))$ minus $((\text{Number A}) - (\text{Number B}))$ equals 2 times Number B, which is a non-negative value.

Given that $((\text{Number A}) - (\text{Number B}))$ and $((\text{Number A}) + (\text{Number B}))$ are 1 and 73, and that $((\text{Number A}) + (\text{Number B}))$ must be greater than or equal to $((\text{Number A}) - (\text{Number B}))$, it logically follows that: $((\text{Number A}) - (\text{Number B})) = 1$ and $((\text{Number A}) + (\text{Number B})) = 73$.

Now we have a system of two simple equations involving Number A and Number B: 1. Number A - Number B = 1 2. Number A + Number B = 73

We can solve this system to find the values of Number A and Number B. A straightforward way is to add the two equations together. When we add Equation 1 and Equation 2, the "Number B" terms cancel out: $(\text{Number A} - \text{Number B}) + (\text{Number A} + \text{Number B}) = 1 + 73$ $\text{Number A} - \text{Number B} + \text{Number A} + \text{Number B} = 74$ $2 \times (\text{Number A}) = 74$

Now, we find Number A by dividing 74 by 2: $\text{Number A} = 74 / 2$ $\text{Number A} = 37$

Now that we know Number A is 37, we can substitute this value back into either of the original simple equations. Let's use the second equation: $37 + \text{Number B} = 73$

To find Number B, we subtract 37 from 73: $\text{Number B} = 73 - 37$ $\text{Number B} = 36$

So, the two numbers are 37 and 36.

... (Verification step)

Final answer: The two numbers are 37 and 36.

1258 L Recurring Error Patterns in OpenAI o3 and Prompt-Based Remedies

1259 OpenAI o3 frequently mirrors the “hand-waving” shortcuts that humans sometimes slip into
1260 proofs—skipping essential justifications and filling the gaps with confident but hollow phrases.
1261 Below we summarize three prominent patterns. We also present prompt instructions that mitigate
1262 each, steering the model toward fuller justifications and greater rigor. However, these prompt
1263 instructions do not necessarily lead to correct solutions. This study is conducted on 10 examples in
1264 the Math set and 10 examples in the Logic set where the model has made the specific mistakes.

1265 **Declaring a result “well-known”.** A favorite pseudo-justification is to appeal to “standard” or
1266 “classical” literature. We counter this by adding: *“This puzzle is novel and has no outside literature
1267 or established solution. Do not cite ‘references’ or ‘classical’ approaches.”* This warning sharply
1268 reduces—but does not entirely eliminate—spurious citations such as “shortlists for problems like
1269 these.”

1270 **Superficial uniqueness proofs.** The model often claims “detailed checking shows...” to assert
1271 uniqueness, even when counterexamples exist. We therefore require: *“When proving uniqueness,
1272 avoid phrases like ‘detailed checking shows’. Explicitly enumerate and rule out every alternative
1273 arrangement.”* While this forces the model to justify each step, it can also encourage unnecessary
1274 brute-force enumeration.

1275 **“Forcing” an answer.** The model sometimes rushes to deliver any construction—correct or
1276 not—rather than pause for rigor. We instruct: *“Prioritize correctness over speed. If uncertain,
1277 admit you are temporarily stuck; feel free to add thinking phases before producing a final answer.”*
1278 This reduces confident but faulty outputs, though it does not inherently expand the range of problems
1279 the model can solve.