

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 REASONING MODEL IS STUBBORN: DIAGNOSING IN- STRUCTION OVERRIDING IN REASONING MODELS

Anonymous authors

Paper under double-blind review

ABSTRACT

Large language models have demonstrated remarkable proficiency in long and complex reasoning tasks. However, they frequently exhibit a problematic reliance on familiar reasoning patterns, a phenomenon we term *reasoning rigidity*. Despite explicit instructions from users, these models often override clearly stated conditions and default to habitual reasoning trajectories, leading to incorrect conclusions. This behavior presents significant challenges, particularly in domains such as mathematics and logic puzzle, where precise adherence to specified constraints is critical. To systematically investigate reasoning rigidity, a behavior largely unexplored in prior work, we introduce a expert-curated diagnostic set, ReasoningTrap. Our dataset includes specially modified variants of existing mathematical benchmarks, namely AIME and MATH500, as well as well-known puzzles deliberately redesigned to require deviation from familiar reasoning strategies. Using this dataset, we identify recurring contamination patterns that occur when models default to ingrained reasoning. We categorize rigidity patterns into three distinctive modes: (i) Interpretation Overload, (ii) Input Distrust, and (iii) Partial Instruction Attention, each causing models to ignore or distort provided instructions. We will publicly release our diagnostic set to facilitate future research on mitigating reasoning rigidity in language models.

1 INTRODUCTION

Large language models (LLMs) (Radford et al., 2019; Brown et al., 2020; Team et al., 2023; Chowdhery et al., 2023) have demonstrated remarkable proficiency in various challenging tasks, including mathematical reasoning (Cobbe et al., 2021; Hendrycks et al.), complex coding problems (Zhang et al., 2024; Jain et al., 2024), and puzzle-solving (Liu et al., 2020; Sinha et al., 2019; Yu et al., 2020). Recently, reasoning models (Jaech et al., 2024; Guo et al., 2025; Team et al., 2025; Team, 2025c; Claude, 2024; Google DeepMind, 2025a) utilizing increased test-time compute have attracted significant attention due to their capability to solve intricate reasoning problems.

However, we pinpoint a problematic behavior from reasoning models, termed *reasoning rigidity*. Crucially, unlike hallucination or a memorization problem, reasoning rigidity reflects a cognitive bias: even when the conditions are fully understood, the model will override them in favor of familiar solution templates. This distinction highlights reasoning rigidity as a unique failure mode that cannot be categorized as an existing problem.

Alarmingly, this reasoning rigidity manifests itself by causing models to override explicit user instructions. As illustrated in Figure 1(a), despite the clear instruction specifying that z is a ‘real number,’ advanced reasoning models capable of solving complex mathematical problems incorrectly assume z must be a ‘complex number’. Similar issues also appear in puzzle contexts; for instance, the explicitly stated condition ‘permanently infertile’ is arbitrarily altered by the model into ‘temporarily infertile,’ thus converting the problem into a familiar Fibonacci sequence scenario. Additionally, direct instructions explicitly stating ‘this is not a Tower of Hanoi problem’ are mistakenly interpreted by the model as a typo, causing it to default to the familiar Tower of Hanoi reasoning. These examples collectively illustrate how LLMs systematically disregard explicit instructions when such directives conflict with their ingrained reasoning patterns.

This rigidity poses challenges across domains where following user-stated constraints is crucial, such as mathematics and logical reasoning that come with multiple conditions that must be ful-

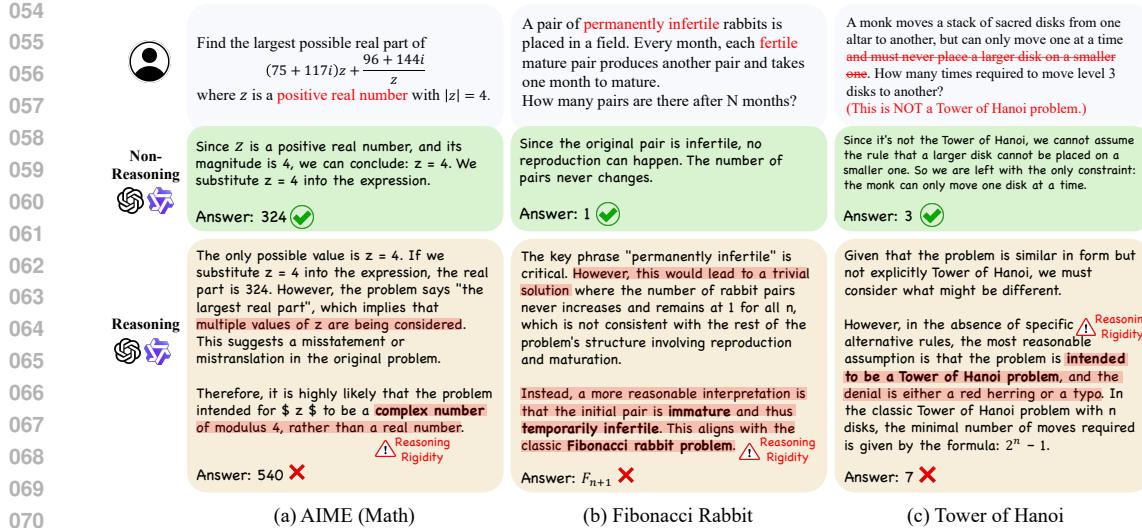


Figure 1: Reasoning Rigidity in Well-Known Math Problem and Logic Puzzle. When solving a subtly modified version of a well-known math problems (AIME) and famous logic puzzles (Fibonacci Rabbit and Tower of Hanoi), advanced reasoning models such as Qwen3-32B and OpenAI o3 default to familiar reasoning template leading to incorrect conclusions.

filled. Through the model’s reasoning rigidity that edits or ignores essential user given conditions, the model’s entire reasoning path can become contaminated by *ingrained reasoning patterns*, ultimately leading to erroneous conclusions or suboptimal solutions. This behavior is highly alarming, but yet to be analyzed to the best of our knowledge. Therefore, there is a need for the evaluation dataset that tackles the reasoning model ability to *faithfully solve the problem within the given condition*, overcoming its innate rigidity to ingrained reasoning patterns introducing contamination to reasoning path.

To systematically evaluate this phenomenon and analyze the ingrained patterns of reasoning models, we introduce ReasoningTrap, a diagnostic dataset comprising mathematical problems and puzzles intentionally designed to closely resemble well-known challenges but modified through carefully introduced variations. ReasoningTrap assesses not only the ability of large language models to detect and incorporate these constraints but also investigates whether these models persistently default to familiar reasoning paths. This diagnostic set thus provides novel insights into both the capabilities and limitations of contemporary reasoning models.

Our analysis of ReasoningTrap yields several important findings: i) reasoning rigidity emerges from unseen training dataset indicating that it is not a simple memorization problem from data overfitting, and ii) such contamination manifests in identifiable, recurring patterns in the models’ outputs. Based on these observations, we propose a budget forcing and prompt hinting to mitigate reasoning rigidity, defined from three distinct rigidity patterns: (i) Interpretation Overload, (ii) Input Distrust, and (iii) Partial Instruction Attention.

Our contributions are as follows:

- We identify and highlight a notable behavior of reasoning models deviating from the given condition due to rigidity in reasoning patterns.
- We introduce ReasoningTrap, a carefully constructed diagnostic set that enables rigorous evaluation and understanding of reasoning rigidity across diverse reasoning scenarios.
- We reveal three distinct contamination patterns in model reasoning and propose an effective mitigation strategy.

Failure Mode	Ability Present	Active Denial	Fixed Reasoning	Primary Symptom
Instruction Following	✗	✗	✗	Cannot follow CoT <i>format</i> directives
Reasoning Faithfulness / Inability	✗	✗	✓	Incorrect / faithless reasoning
Memorization	✗	✗	✓	Failure to generalize
Overthinking	✓	△	△	Unnecessarily long CoT
Reasoning Rigidity	✓	✓	✓ (content)	Refuses the given conditions to follow familiar reasoning

Table 1: Condensed comparison of reasoning-related failure modes. Legend: \checkmark = strongly related; \triangle = partial; \times = unrelated.

2 RELATED WORKS

Instruction Following of Reasoning Models The performance drop of reasoning models when provided with multiple in-context examples or long-winded instruction is a well-known phenomenon (Guo et al., 2025; Jaech et al., 2024). Such phenomenon states that reasoning models are less capable of following user-provided examples. Our work investigates the phenomenon that reasoning models are capable of following instructions from the user, but sticks to the familiar reasoning pattern thus conform less to the given instruction.

Memorization. Memorization problem occurs when models rely on instance-specific patterns seen during training and do not have the ability to generalize to novel or structurally altered problems. In contrast, reasoning rigidity arises even when the model possesses strong ability in understanding and solving the problem, but nevertheless defaults to a familiar reasoning template rather than adapting its reasoning strategy.

Rigidity in Reasoning Models Several works have pointed out the possibility that LLM models show rigid pattern in reasoning in specific subfields, medical domain (Kim et al., 2025) and educational domain (Araya, 2025). Our work is the first to systematically analyze the reasoning rigidity in larger domain including mathematics and puzzles.

Closely related to our work, are several previous studies that explore creativity and generalizability in large language models (LLMs). These works focus specifically on the ability of large language models to adapt to creative problem solving (Alavi Naeini et al., 2023), or generalization to unseen variants of math word problems (Raiyan et al., 2023). On the other hand, our work specifically examines the underlying model-driven rigidity of reasoning models, and identifying deliberate overrides of given problem conditions *rather than mere inability to solve tasks creatively or generalizing*.

As highlighted in Table 1, our work differs from these works since the focus is on the ingrained reasoning rigidity pattern, *rather than mere inability to solve tasks, follow format level instructions, or diversify reasoning direction or style*. For detailed explanation on how reasoning rigidity differs from other failure modes, please refer to Section A.

3 ANALYSIS ON REASONING RIGIDITY PHENOMENON

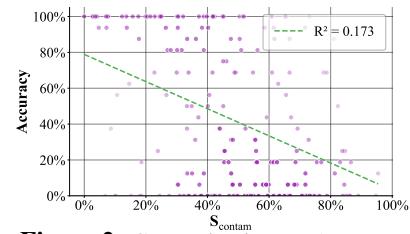
Definition. Reasoning rigidity refers to a phenomenon in which a model correctly *understands* a problem and possesses the capability to solve it, yet *fails to solve the problem because rigidly defaults to ingrained reasoning patterns*. It manifests by the models’ behavior overriding or ignoring the given conditions in the problem.

Structure of Diagnostic Set. Existing reasoning datasets cannot diagnose rigidity because it requires observing the model’s behavior to default to familiar but wrong reasoning trace. Therefore, it requires modified problems whose solutions diverge from the originally familiar problems. In this section, we first investigate the symptom of reasoning rigidity, by quantifying the extent to which reasoning on modified problem is contaminated by wrong but familiar original problems through our diagnostic set `ReasoningTrap`.

Our diagnostic set consists of original question-solution-answer triplets $(q_{\text{orig}}, s_{\text{orig}}, a_{\text{orig}})$ and their modified counterparts $(q_{\text{mod}}, s_{\text{mod}}, a_{\text{mod}})$. The modified solutions and answers diverge from the original to assess if the reasoning correctly follows the instructions stated in the modified question,

162 **Table 2:** Contamination scores for open-source base vs. reasoning
 163 models.

Model Pair	AIME		MATH500	
	Base	Reason	Base	Reason
DeepSeek V3 vs. R1	44.6%	45.1%	51.1%	55.9%
Qwen2.5 32B vs. QwQ	42.9%	49.6%	47.3%	50.7%
Qwen3 32B No Think vs. Think	46.8%	51.7%	48.7%	55.0%



170 **Figure 2:** Contamination vs. Accuracy

171 not the original one. The examples of original and modified triplets are shown in Section C, and the
 172 dataset construction process is detailed in Section 4.

173 3.1 CONTAMINATION SCORE: INDICATOR OF REASONING RIGIDITY

175 Given a model f , we denote the reasoning traces as $\hat{r}_{\text{orig}} = f(\mathbf{q}_{\text{orig}})$ for original questions and
 176 $\hat{r}_{\text{mod}} = f(\mathbf{q}_{\text{mod}})$ for modified questions. A reasoning trace for modified question is considered
 177 **contaminated** on a paired item when its reasoning is highly similar to the reasoning used for the
 178 original question, despite the modified problem requiring a different approach.

179 We quantify the contamination in the model’s reasoning trace as

$$181 \quad S_{\text{contam}} = \frac{1}{p} \sum_{i=1}^p \mathbb{1}[\text{cs}(\mathbf{s}_{\text{orig}}, \hat{r}_{\text{mod}}^i) > \text{cs}(\mathbf{s}_{\text{mod}}, \hat{r}_{\text{mod}}^i)],$$

184 where $\text{cs}(\cdot, \cdot)$ denotes cosine similarity between p embedded reasoning blocks
 185 $[\hat{r}_{\text{mod}}^1, \hat{r}_{\text{mod}}^2, \dots, \hat{r}_{\text{mod}}^p]$ ¹ and the ground truth solutions \mathbf{s}_{orig} or \mathbf{s}_{mod} . Note that if the model
 186 do not have rigidity, then most of the model reasoning trace \hat{r}_{mod} should be closer to the ground
 187 truth solution \mathbf{s}_{mod} , not \mathbf{s}_{orig} . Therefore, a higher contamination score suggests that the model
 188 tends to defaults to familiar reasoning templates instead of incorporating the modified constraints.

189 **Contamination Comparison in Base vs Reasoning Models.** The contamination scores in Ta-
 190 ble 2 indicate that reasoning-oriented models generate a larger fraction of reasoning blocks that
 191 more closely align with the solution pattern of the *original* question \mathbf{s}_{orig} , rather than with the
 192 correct reasoning required by the *modified* problem \mathbf{s}_{mod} . This pattern suggests that reasoning models,
 193 despite having stronger overall reasoning ability, are more likely to default to familiar solution tra-
 194 jectories when confronted with modified variant to familiar original questions.

195 **Contamination Score Reflect Reasoning Rigidity Only Partially** Contamination score shows
 196 whether the reasoning trace for *modified* question shares similarity with reasoning on *original* ques-
 197 tion. However, contamination score emits a dual signal, one is a legitimate reasoning behavior that
 198 explores diverse reasoning approach, and the second is a malignant behavior that gets trapped into a
 199 familiar yet wrong reasoning trace. The dual signal is shown in Figure 2, high contamination scores
 200 often correspond to incorrect reasoning and lower accuracy, but in some cases they coincide with
 201 correct reasoning and high accuracy, therefore contamination score implies reasoning rigidity, but
 202 not is a direct measurement².

204 3.2 REASONING RIGIDITY: HIGH PERCEPTION BUT LOW ACCURACY

206 **Perception and Pass@1 Score.** To directly show reasoning rigidity, it is important to determine
 207 whether the model accurately perceives the given problem and tries to solve in a correct direction,
 208 but eventually falls into a wrong answer. The perception score measures whether the model correctly
 209 understands the problem setup and tries to solve as instructed by the question by comparing with
 210 the given question and ground truth solution using an auxiliary LLM judges. For this purpose, gpt-
 211 4o-mini is prompted with a instruction to determine whether a subset of model reasoning is similar
 212 with the ground truth. The full prompt is shown in Section D.1.

213 ¹Reasoning blocks are paragraphs that are split by double line breaks and encoded using OpenAI’s text-
 214 embedding-small model.

215 ²Each point corresponds to a specific model-problem pair, covering six models (DeepSeek V3, R1, Qwen2.5
 32B, QwQ, and Qwen3 32B Think and No-Think) evaluated on AIME and MATH500 problems.

216 **Table 3: Comparison of Base vs. Reasoning Models on ConditionedMath and**
 217 **PuzzleTrivial.** *OpenAI o3-mini and o4-mini do not open reasoning trajectory to users, there-
 218 **fore we are unable to measure perception score (p-score) for these two models.**

Model Name	Type	AIME		MATH500		PuzzleTrivial	
		pass@1	p-score	pass@1	p-score	pass@1	p-score
Qwen2.5-32B-Instruct	Base	45.77 \pm 7.22	75.55 \pm 5.01	40.88 \pm 5.74	70.37 \pm 4.39	30.23 \pm 3.51	72.97 \pm 3.01
+ QwQ-32B	Reason	42.46 \pm 6.63	81.80 \pm 4.27	34.75 \pm 5.74	71.37 \pm 4.59	38.36 \pm 4.38	97.66 \pm 0.48
Qwen3-32B No think	Base	40.07 \pm 6.68	90.81 \pm 2.66	43.75 \pm 5.59	85.88 \pm 2.90	65.55 \pm 3.20	84.21 \pm 2.07
+ Qwen3-32B Think	Reason	29.60 \pm 6.32	76.84 \pm 4.91	30.63 \pm 5.59	75.50 \pm 3.74	37.19 \pm 3.40	96.33 \pm 0.64
Qwen3-235B No think	Base	40.99 \pm 7.04	86.40 \pm 3.08	46.50 \pm 5.34	84.25 \pm 2.72	59.84 \pm 3.38	86.17 \pm 2.66
+ Qwen3-235B Think	Reason	20.77 \pm 5.07	81.62 \pm 4.12	23.25 \pm 4.63	79.13 \pm 3.39	37.97 \pm 4.05	97.42 \pm 0.56
DeepSeek V3	Base	48.35 \pm 6.82	77.94 \pm 5.46	42.50 \pm 5.25	75.00 \pm 4.57	64.45 \pm 4.28	80.00 \pm 3.45
+ DeepSeek R1	Reason	39.71 \pm 7.76	80.88 \pm 5.07	38.00 \pm 6.40	73.00 \pm 5.09	50.55 \pm 4.33	97.27 \pm 0.97
EXAONE 4.0 32B No think	Base	43.01 \pm 6.71	84.01 \pm 3.55	33.12 \pm 5.66	70.00 \pm 4.10	41.25 \pm 3.78	81.72 \pm 2.80
+ EXAONE 4.0 32B Think	Reason	20.22 \pm 4.52	77.76 \pm 4.86	24.62 \pm 5.10	70.12 \pm 4.09	34.45 \pm 3.32	85.00 \pm 2.03
Nemotron Nano 9B v2 No think	Base	51.84 \pm 7.80	80.51 \pm 5.53	39.75 \pm 5.56	81.62 \pm 3.77	42.42 \pm 4.48	85.08 \pm 2.65
+ Nemotron Nano 9B v2 Think	Reason	47.79 \pm 6.82	78.86 \pm 4.94	36.12 \pm 5.42	78.25 \pm 3.80	42.34 \pm 4.35	83.52 \pm 2.88
GPT-4o	Base	47.06 \pm 7.06	82.35 \pm 3.54	35.50 \pm 4.89	69.87 \pm 3.93	48.38 \pm 4.53	75.23 \pm 3.63
ChatGPT-4o	Base	33.82 \pm 6.99	84.56 \pm 4.35	38.00 \pm 3.26	81.50 \pm 3.26	58.59 \pm 3.63	89.14 \pm 2.18
+ o3-mini	Reason	22.79 \pm 5.72	.	38.00 \pm 5.81	.	39.22 \pm 4.49	.
+ o4-mini	Reason	19.12 \pm 5.49	.	26.50 \pm 5.17	.	29.53 \pm 4.18	.
Gemini2.5 Flash No think	Base	52.21 \pm 7.17	82.35 \pm 3.54	49.80 \pm 5.59	69.87 \pm 3.93	65.94 \pm 4.27	94.06 \pm 1.79
+ Gemini2.5 Flash Think	Reason	46.12 \pm 7.33	89.81 \pm 2.52	47.95 \pm 6.27	82.51 \pm 3.47	65.63 \pm 4.34	94.06 \pm 1.95
Claude 3.7 Sonnet No think	Base	50.74 \pm 7.65	80.15 \pm 4.94	36.00 \pm 5.49	85.50 \pm 2.95	73.28 \pm 4.03	89.30 \pm 2.05
+ Claude 3.7 Sonnet Think	Reason	46.72 \pm 7.63	72.99 \pm 6.01	32.00 \pm 5.58	78.00 \pm 4.44	52.81 \pm 4.58	79.69 \pm 3.50

237
 238 Formally, perception score is defined as

$$\text{p-score} = \frac{1}{N} \sum_{i=1}^N p_i,$$

242 where $p_i \in \{0, 1\}$ indicates whether the model’s reasoning correctly incorporates the problem con-
 243 ditions. A high perception score indicates that the model understands the modified problem.

244 When such understanding is accompanied by a degradation in Pass@1 performance, this pattern
 245 provides evidence of reasoning rigidity that the model comprehends the task but fails to adjust its
 246 reasoning strategy, consistent with a fallback to familiar reasoning templates.

248 **Rigidity Patterns Emerge in Various LRM_s, from Open-sourced to Proprietary Models.**

249 Across most configurations, reasoning-oriented models underperform their base-model counterparts
 250 on both ConditionedMath and PuzzleTrivial, as shown in Table 3. This result is counter-
 251 intuitive given their superior performance on standard reasoning benchmarks.

252 Reasoning models exhibit *higher* perception score (reason: 83.66 vs. base: 81.35), indicating that
 253 they correctly interpret the modified questions more often than base models. Yet despite this superior
 254 perception, their Pass@1 performance remains lower. This divergence between understanding and
 255 execution provides further evidence of reasoning rigidity: reasoning models accurately comprehend
 256 the task but fail to adapt their reasoning strategy, defaulting instead to familiar solution templates.

258 **Reasoning Rigidity Appears Across All Difficulty Levels.** Our diagnostic set,
 259 ReasoningTrap, consists of intermediate level to olympiad level problems, as measured
 260 following the protocol of SkyT1 (Team, 2025a), which tags AoPS difficulty levels using the official
 261 template. From Figure 3, we observe that reasoning rigidity is universally observed over all
 262 difficulty levels, not only in easy tasks. Moreover, the difficulty distributions of the original and
 263 modified problems maintains similar level of task difficulty as shown in the histogram.

264 **Task Familiarity leads to Reasoning Rigidity.** We next examine whether reasoning rigidity per-
 265 sists in *out-of-domain* (OOD) settings, where models are unlikely to have encountered similar prob-
 266 lems during post-training. If rigidity arises primarily from task-level familiarity, rather than instance-
 267 level memorization, we should observe similar rigidity from unseen problems.

268 We evaluate four reasoning vs. base model pairs (Qwen2.5-32B vs. QwQ-32B, Qwen3-32B
 269 No-Think vs. Think, DeepSeek V3 vs. R1, and Claude 3.7 Sonnet No-Think vs. Think) on the

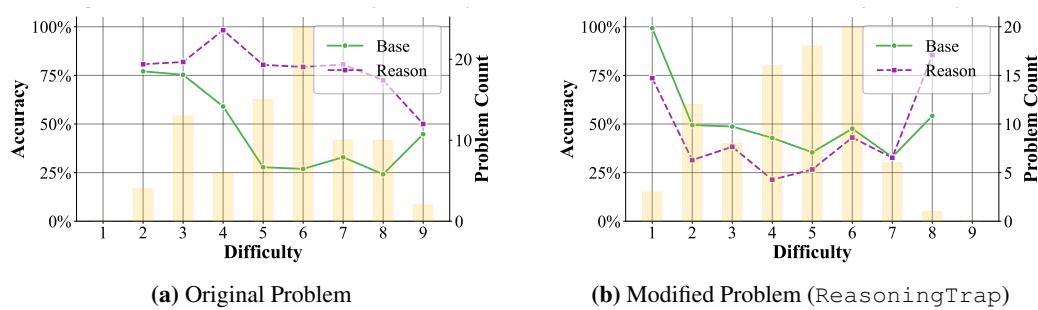


Figure 3: In original MATH500 and AIME problems, reasoning models outperforms base models over all difficulty levels. On the other hand, our dataset ReasoningTrap shows a reversed trend where base models outperform reasoning models even at the same difficulty level. The difficulty levels are measured per question by AoPS standard.

ProofWriter dataset (Tafjord et al., 2021) at depth 3 under the closed-world assumption, where the correct answer is always true. To create modified instances, we remove two conditions and two rules from each problem, breaking the deductive reasoning chain. In these cases, the logically correct answer becomes either unknown or false.

Table 4: Base vs. Reasoning model performance in the OOD domain.

Model	Modified		Original	
	Base	Reason	Base	Reason
Qwen2.5-32B vs. QwQ-32B	28%	34%	87%	95%
Qwen3-32B No-Think vs. Think	33%	33%	97%	99%
DeepSeek V3 vs. R1	30%	35%	99%	100%
Claude 3.7 Sonnet No-Think vs. Think	28%	29%	100%	100%

In Table 4, we demonstrate that reasoning models do not perform worse than their base model counterparts on the OOD modified tasks. Combined with the fact that reasoning models retain their strong performance on the original ProofWriter items, these results indicate that rigidity does *not* automatically arise in unseen domains. Instead, rigidity appears to rely on *task-level familiarity*, emerging when models have learned strong prior reasoning templates. In unfamiliar OOD settings, such templates are absent, and rigidity effects are hardly observed.

Reasoning Rigidity Persists Beyond Memorization Effects. We investigate whether reasoning rigidity arises from *task-level familiarity*, rather than instance-level memorization. we test whether rigidity also appears on *unseen* mathematical problems that models could not have memorized. We evaluate DeepSeek-R1 and DeepSeek-V3 on AIME 2025 problems, which were released after both models were trained. We modify 9 AIME25 questions to create paired variants original and modified question-solution-answer pairs and sample 16 responses per problem.

Table 5: Pass@1 Score of Base vs Reasoning Models in Seen / Unseen Train Data

Train Time	Base (V3)	Reason (R1)
Seen	45.59	39.71
Unseen	52.78	41.67

The results in Table 5 show that the reasoning model DeepSeek-R1 underperforms its base-model counterpart DeepSeek-V3 on unseen AIME 2025 questions, despite R1 being substantially stronger on standard mathematical benchmarks.

Because these problems were released after both models' training cutoffs, the presence of rigidity on truly novel items rules out instance-level memorization as an explanation. Moreover, the pass@1 performance on seen and unseen problems is comparable, suggesting that rigidity arises from task-level familiarity rather than from memorizing specific training examples. Taken together, these results indicate that reasoning rigidity reflects a deeper limitation: the model's tendency to revert to familiar reasoning templates even when the problem requires a different solution strategy.

324
325
Table 6: Analysis on the Effect of Budget Forcing.

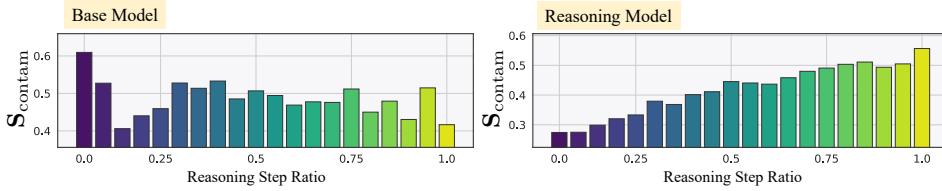
326	327	328	Model	ConditionedMath	Original	ConditionedMath	Original																																									
329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	373	374	375	376	377
			Qwen3-32B	ConditionedMath AIME	Original AIME	ConditionedMath MATH500	Original MATH500																																									
				pass@1	pass@1	pass@1	pass@1																																									
			Budget + low	29.60 \pm 6.32	72.79 \pm 6.95	30.63 \pm 5.59	85.50 \pm 4.69																																									
			Budget + medium	51.47 \pm 7.46	28.68 \pm 5.98	42.00 \pm 5.91	68.00 \pm 5.39																																									
			Budget + high	39.71 \pm 6.69	50.00 \pm 7.76	36.00 \pm 5.90	76.50 \pm 5.32																																									
				36.03 \pm 6.94	57.35 \pm 7.35	34.00 \pm 5.92	81.00 \pm 5.13																																									

334 3.3 HYPOTHESIS ON THE REASONING RIGIDITY CAUSES

335 We suggest three suspects that lead to reasoning rigidity and test whether each factor has a possibility
 336 to mitigate rigidity problems.

339 **Model Output Length and Reasoning Rigidity.** We hypothesize that the overthinking behavior
 340 is one of the cause of degraded performance in ReasoningTrap. Following Team (2025b), we
 341 append the prompt ‘Considering the limited time by the user, I have to give the solution based on
 342 the thinking directly now. </think>’ to the generated response, instructing the model to produce
 343 an answer once a predefined token budget is reached. We apply token budgets for each dataset and
 344 report pass@1 scores. For MATH500, the budgets are 2k, 4k, and 6k tokens; for AIME which is
 345 more challenging, we use 2k, 6k, and 10k tokens.

346 Motivated by the observation from Figure 4 that contamination scores increase in the later stages
 347 of a model’s reasoning trace, we impose a token budget to halt the model before it enters familiar
 348 but incorrect reasoning patterns caused by rigidity. As shown in Table 6, reducing the token
 349 budget improves performance on our diagnostic set but simultaneously degrades performance on
 350 the original benchmarks. This trade-off indicates that strict budget forcing suppresses both produc-
 351 tive exploratory reasoning and harmful rigid reasoning, revealing a fundamental limitation of this
 352 intervention.

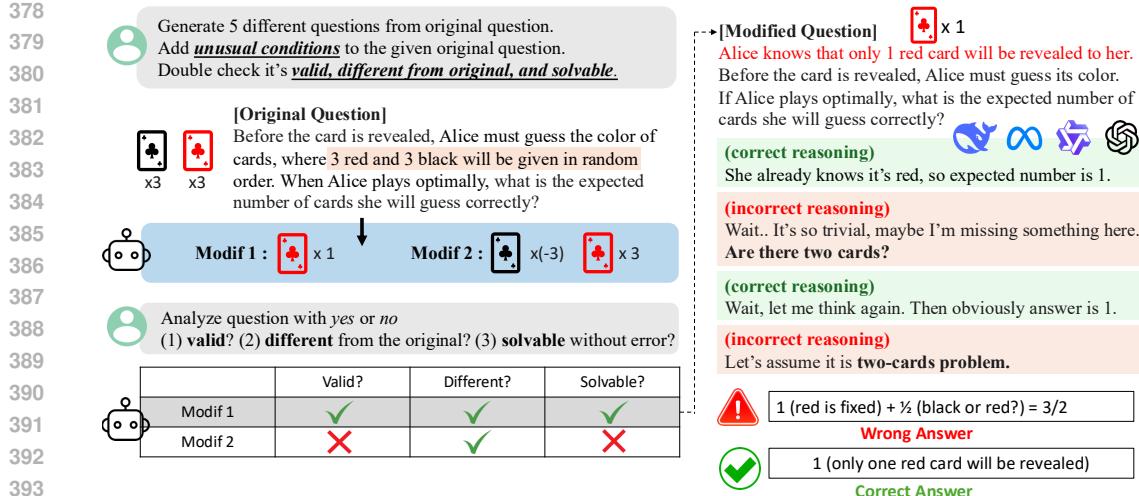
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
Figure 4: Contamination Score and Token Length

362 **Model Entropy Correlates with Rigidity Patterns** We test the hypothesis that RL training inten-
 363 sifies rigidity patterns in language models, and conduct a preliminary experiment to validate this by
 364 comparing the entropy of ReasoningTrap (modified AIME), and the original AIME dataset.

365 We compute next-token entropy defined as $H_t = -\sum p_\theta(v | x_{1:t-1}) \log p_\theta(v | x_{1:t-1})$, averaged
 366 over all positions and prompts in the dataset. The model is Qwen2.5 7B base (before RL), and
 367 the same model trained with DAPO algorithm (Open Zero Reasoner dataset) for the stated training
 368 steps. As shown in the Table 7, the increase of entropy in the original dataset is explosive, whereas
 369 the increase in entropy in our dataset is comparably limited (note that lower entropy indicates model
 370 output rigidity).

371
372 **Table 7: Entropy on Original vs. Modified AIME.** The percentage in parenthesis indicates the
 373 increase in entropy compared to the previous iteration.

374	Dataset	Base Model	Iter 120	Iter 260
375	Original AIME (22–24)	0.14	1.36 (+871%)	2.47 (+81.6%)
376	Modified AIME (ReasoningTrap)	0.34	1.25 (+268%)	1.72 (+37.6%)



395 **Figure 5: Dataset Construction Pipeline** The dataset construction pipeline of
 396 ConditionedMath first creates new questions with unfamiliar conditions and verify whether
 397 the modified questions are (1) valid, (2) different, and (3) solvable. Among Modif 1 and Modif 2,
 398 the latter is omitted as it includes an invalid condition of multiplying by -3 . When solving Modif
 399 1, reasoning models over-complicate the problem, overriding the simple logic by defaulting to a
 400 familiar logic of assuming a two-card setup.

402 Both datasets (Original and ReasoningTrap) are evaluated on the identical model, before and
 403 after RL training, so any change in entropy must be attributed to the subsequent RL updates. Because
 404 the input distribution is constant, the only changing component is the model. The entropy experiment
 405 indicates a model–dataset interaction effect induced by RL. This observation supports the hypothesis
 406 that RL training differently affects the output distribution of the original mathematical dataset and
 407 our dataset.

4 REASONINGTRAP: REASONING RIGIDITY DIAGNOSTIC SET

411 In this section, we introduce ReasoningTrap, a well-curated diagnostic set specifically designed
 412 to reveal reasoning rigidity in language models.

414 **ConditionedMath.** To systematically investigate this phenomenon, we curate two specialized
 415 datasets: ConditionedMath which consists of challenging mathematical problems augmented
 416 with novel constraints, and PuzzleTrivial comprised of puzzle questions that are subtly modified
 417 version from logic puzzles. We construct the ConditionedMath dataset by adapting ques-
 418 tions from historical AIME 2022–2024 (AIME) and MATH500 Level 5 (Hendrycks et al.) datasets.

420 **PuzzleTrivial.** We develop PuzzleTrivial dataset from a classic puzzle questions by
 421 subtly modifying premises or omitting specific constraints, thereby altering the logical reasoning
 422 required, inspired by Williams & Huckle (2024); Vellum AI (2025). PuzzleTrivial spans eight
 423 unique puzzle themes, and proliferated into 10 versions of logic puzzles for each theme. We ensure
 424 that ambiguity do not exist by generating clarifying instructions such as ‘find the simplest valid
 425 solution’.

426 Since mathematical reasoning task, ConditionedMath, requires modification to logic-sensitive
 427 mathematical conditions, the construction and verification process should be meticulously moni-
 428 tored. To this end, we introduce a two-stage construction and verification pipeline in the following
 429 sections. The construction follows the pipeline consists of two stages, (1) original question modi-
 430 fication in Section 4.1, and (2) dataset filtering in Section 4.2 as detailed in Figure 5. Note that
 431 PuzzleTrivial starts from a human generated modification, and logic-invariant conditions are
 432 altered to proliferate the number of questions.

432 4.1 ORIGINAL QUESTION MODIFICATION
433434 We use OpenAI gpt-4o-mini for LLM-based dataset generation and o4-mini for LLM-based dataset
435 filtering, since filtering requires a more powerful language model as a verifier. After LLM-based
436 filtering, the dataset are further filtered into smaller high-quality subset by human annotators.
437438 For generating novel conditions, we prompt the model to modify the original problem and solution
439 into five distinct versions that meaningfully alter the problem’s reasoning trajectory and eventually
440 lead to a different answer. The prompt to modify the original question is provided in Section B.1.
441 As this process proliferates the original dataset into five-fold larger dataset with unverified modifi-
442 cations, we proceed to dataset filtering process that incorporates monitoring from both strong LLM
443 and human expert.
444445 4.2 DATASET FILTERING PIPELINE
446447 **LLM Verification.** These modified questions are further validated on three critical criteria with
448 OpenAI o4-mini model: (a) mathematical validity of the modified conditions to ensure that no
449 internal contradictions exist, (b) divergence of the resulting solution from the original problem’s
450 solution, and (c) existence of solution. The final criterion is to facilitate the assessment on whether
451 the model continues to employ its previously learned reasoning paths or effectively generates a new
452 reasoning trajectory as dictated by the modified conditions.
453454 **Human Verification.** Three annotators with mathematical expertise evaluated each ques-
455 tion–solution pair using three criteria - valid, diverge, and solvable - each rated on a five-point
456 Likert scale (1 = negative, 5 = positive). For each pair, the annotators recorded a score for all three
457 criteria. We retained only those questions receiving a score of 5 on all three dimensions, ensuring
458 an expert-quality dataset.459 During filtering, the AIME dataset’s 90 origi-
460 nal question–answer pairs were expanded into
461 five variants each (450 total), from which 34
462 questions satisfied all validity criteria. Sim-
463ilarly, the 130 Level-5 problems from the
464 MATH500 dataset were expanded into 650
465 variants, yielding 50 validated items after fil-
466 tering. Puzzle dataset starts from 8 completely
467 distinct themes, proliferated into 160 ques-
468 tions, and filtered down into 80 questions.
469470 The quality of the dataset and the LLM-based verification procedure was assessed by human eval-
471 uators and reported in Section E to ensure the reliability of our construction and filtering process.
472473 5 SIMPLE REASONING RIGIDITY MITIGATION IN INFERENCE LEVEL
474475 Across various reasoning models, we identify three universal patterns when reasoning rigidity
476 emerges. The provided taxonomy of reasoning rigidity, illustrate in Figure 6, is stated as follows.477 **Interpretation Overload** The model starts to reject the given question conditions by reinterpreting
478 the question into multiple ways rather than accepting a straightforward interpretation.479 **Input Distrust** Reasoning models have a unique patterns assuming the presence of typos, translation
480 mistake, or input errors. This leads to the dismissal of the conditions stated in the question and make
481 the reasoning process overly complicated even in the straightforward cases.482 **Partial Instruction Attention** The models focus selectively on a portion of provided instructions,
483 typically to the latter or more salient part.484 Using o4-mini to detect the most dominant patterns in each trajectory, we count their total occur-
485 rences for each model and report the percentage share of each pattern in Table 9.453 **Table 8: Diagnostic Dataset Configuration**

	ConditionedMath	PuzzleTrivial
	AIME	MATH500
Original Size	90	130
Before Filtering	450	650
After Filtering	34	50
		8 Themes
		N/A
		N/A

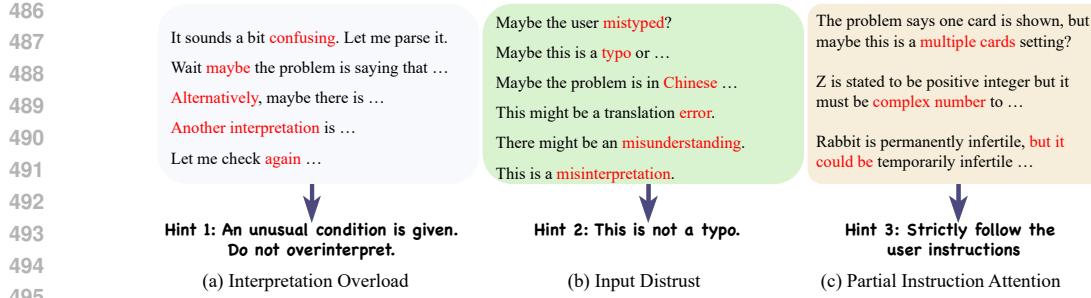


Figure 6: Three Rrigidity Patterns.

Table 9: Distribution of Rrigidity Error Patterns across Models.

Model	Interpretation Overload	Input Distrust	Partial Attention	Total
Qwen3 (32B)	59.48%	18.95%	21.57%	100%
Claude (3.7 Sonnet)	75.76%	6.06%	18.18%	100%
Gemini 2.5 Pro	61.49%	7.69%	30.82%	100%
Deepseek R1	67.35%	19.39%	13.27%	100%

We analyze how often each rigidity pattern appears in model responses that fall back on familiar but incorrect reasoning in ReasoningTrap. Using o4-mini to detect the most dominant pattern in each trajectory, we count their total occurrences for each model and report the percentage share of each pattern in Table 9.

Among the three types, interpretation overload is the most common across all four reasoning models, especially in Claude 3.7 Sonnet. In contrast, Qwen3 and DeepSeek R1 more often show input distrust, while Gemini 2.5 Pro tends to display partial attention to user instructions.

Based on these observations, we investigate the effectiveness of simple prompt hinting. Following the three patterns that various reasoning models universally share, we introduce an additional prompt to the model’s response, explicitly stating a tailored hint for each pattern as shown in Figure 6. Using pattern-based hinting, we test variants of the additional prompt hints based on the three major patterns. As shown in Table 10, the first hint that remedies interpretation overload shows performance improvement in both original and modified variants of AIME and MATH500. This states that when provided with appropriate instruction, reasoning models robustly solve both familiar reasoning tasks and unfamiliar variants altogether. However, as hinting to strictly follow the user instruction rather drops the pass@1 score for the original AIME, the design of instruction should be meticulously chosen considering the model and dataset type. The model response change from hinting is provided in Section G.4.

Table 10: Prompt Hinting on ReasoningTrap.

	AIME		MATH500	
	ConditionedMath	Original	ConditionedMath	Original
Qwen3-32B	29.60 \pm 6.32	72.79 \pm 6.95	30.63 \pm 5.59	85.50 \pm 4.69
+ Hint 1	42.65 \pm 8.14	75.74 \pm 6.55	40.50 \pm 6.46	85.50 \pm 4.41
+ Hint 2	37.50 \pm 7.48	73.53 \pm 6.15	37.00 \pm 6.20	85.00 \pm 4.63
+ Hint 3	36.03 \pm 7.17	69.85 \pm 6.82	32.00 \pm 5.85	87.00 \pm 4.24

6 CONCLUSION

To the best of our knowledge, this is the first work to reveal the surprising rigidity that advanced reasoning models exhibit during multi-step reasoning. To systematically study this phenomenon, we curate a high-quality diagnostic dataset that measures reasoning rigidity and contamination from memorized solution trajectories. Our analysis shows that rigidity arises even in problems unseen during training, confirming that it is not a simple memorization issue. Beyond diagnosis, we demonstrate that lightweight inference-time strategies can partially alleviate rigidity. However, our findings unveil that the root cause lies in the reinforcement learning-based training process itself. In particular, the train-time entropy patterns we uncover highlight the need to rethink how reasoning models are optimized. We argue that addressing this hidden rigidity is essential for building credible reasoning systems.

540 REFERENCES
541

542 AIME. AIME 2024. https://artofproblemsolving.com/wiki/index.php/2024_AIME_I?srsltid=AfmBOoqfUhmDQZd1-etOmNCjXpUgzyI4604azz8hjLFPLSGMw_35PqJJ. Accessed: 2025-05. 8

545 Saeid Alavi Naeini, Raeid Saqur, Mozhgan Saeidi, John Giorgi, and Babak Taati. Large language
546 models are fixated by red herrings: Exploring creative problem solving and einstellung effect
547 using the only connect wall dataset. *Advances in Neural Information Processing Systems*, 36:
548 5631–5652, 2023. 3, 15

549 Roberto Araya. Do chains-of-thoughts of large language models suffer from hallucinations, cogni-
550 tive biases, or phobias in bayesian reasoning? *arXiv preprint arXiv:2503.15268*, 2025. 3
551

552 Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal,
553 Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are
554 few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

555 Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang,
556 Denghui Zhang, Tong Zhang, et al. Rm-r1: Reward modeling as reasoning. *arXiv preprint*
557 *arXiv:2505.02387*, 2025a. 27

559 Yanda Chen, Joe Benton, Ansh Radhakrishnan, Jonathan Uesato, Carson Denison, John Schulman,
560 Arushi Soman, Peter Hase, Misha Wagner, Fabien Roger, et al. Reasoning models don't always
561 say what they think. *arXiv preprint arXiv:2505.05410*, 2025b. 15

562 Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam
563 Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm:
564 Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):
565 1–113, 2023. 1

566 Claude. Claude 3.5 Sonnet. <https://www.anthropic.com/news/claude-3-5-sonnet>, June 2024. Accessed: 2025-05. 1

569 Claude. Claude 3.7 Sonnet. <https://www.anthropic.com/news/claude-3-7-sonnet>, February 2025. Accessed: 2025-05. 23

571 Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser,
572 Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to
573 solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 1

575 Ganqu Cui, Lifan Yuan, Zefan Wang, Hanbin Wang, Wendi Li, Bingxiang He, Yuchen Fan, Tianyu
576 Yu, Qixin Xu, Weize Chen, et al. Process reinforcement through implicit rewards. *arXiv preprint*
577 *arXiv:2502.01456*, 2025. 27

578 DeepSeek-AI. Deepseek-v3 technical report, 2024. URL <https://arxiv.org/abs/2412.19437>. 23

581 DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning,
582 2025. URL <https://arxiv.org/abs/2501.12948>. 23

583 Hugging Face. Open r1: A fully open reproduction of deepseek-r1, January 2025. URL <https://github.com/huggingface/open-r1>. 26

586 Tingchen Fu, Jiawei Gu, Yafu Li, Xiaoye Qu, and Yu Cheng. Scaling reasoning, losing control:
587 Evaluating instruction following in large reasoning models. *arXiv preprint arXiv:2505.14810*,
588 2025. URL <https://arxiv.org/abs/2505.14810>. 15

589 Google DeepMind. Gemini 2.5 Pro. <https://deepmind.google/technologies/gemini/pro/>, March 2025a. Accessed: 2025-05. 1

592 Google DeepMind. Gemini 2.5 flash: Faster, lower-cost reasoning. <https://blog.google/technology/google-deepmind/google-gemini-updates-io-2025/>, 2025b.
593 Blog post, accessed 22 May 2025. 23

594 Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu,
 595 Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms
 596 via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 1, 3
 597

598 Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn
 599 Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset.
 600 In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks*
 601 *Track (Round 2)*. 1, 8

602 Jingcheng Hu, Yinmin Zhang, Qi Han, Dixin Jiang, Xiangyu Zhang, and Heung-Yeung Shum.
 603 Open-reasoner-zero: An open source approach to scaling up reinforcement learning on the base
 604 model, 2025. URL <https://arxiv.org/abs/2503.24290>. 26, 27

605 Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec
 606 Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv*
 607 *preprint arXiv:2412.16720*, 2024. 1, 3

608 Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando
 609 Solar-Lezama, Koushik Sen, and Ion Stoica. Livecodebench: Holistic and contamination free
 610 evaluation of large language models for code. *arXiv preprint arXiv:2403.07974*, 2024. 1

611 Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjun Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin
 612 Jiang, Qun Liu, and Wei Wang. Followbench: A multi-level fine-grained constraints following
 613 benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the As-
 614 sociation for Computational Linguistics (ACL-2024), Long Papers*, pp. 4667–4688, 2024. URL
 615 <https://aclanthology.org/2024.acl-long.257/>. 15

616 Muhammad Khalifa, Rishabh Agarwal, Lajanugen Logeswaran, Jaekyeom Kim, Hao Peng, Moon-
 617 tae Lee, Honglak Lee, and Lu Wang. Process reward models that think. *arXiv preprint*
 618 *arXiv:2504.16828*, 2025. URL <https://arxiv.org/abs/2504.16828>. 26

619 Jonathan Kim, Anna Podlasek, Kie Shidara, Feng Liu, Ahmed Alaa, and Danilo Bernardo. Lim-
 620 itations of large language models in clinical problem-solving arising from inflexible reasoning.
 621 *arXiv preprint arXiv:2502.04381*, 2025. 3

622 Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. Logiqa: A
 623 challenge dataset for machine reading comprehension with logical reasoning. *arXiv preprint*
 624 *arXiv:2007.08124*, 2020. 1

625 Michael Luo, Sijun Tan, Justin Wong, Xiaoxiang Shi, William Y. Tang, Manan Roongta, Colin Cai,
 626 Jeffrey Luo, Li Erran Li, Raluca Ada Popa, and Ion Stoica. Deepscaler: Surpassing o1-preview
 627 with a 1.5b model by scaling rl, 2025. Notion Blog. 26

628 Andreas Madsen, Sarath Chandar, and Siva Reddy. Are self-explanations from large language mod-
 629 els faithful? In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 295–
 630 337, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/
 631 v1/2024.findings-acl.19. URL [https://aclanthology.org/2024.findings-acl.
 632 19/](https://aclanthology.org/2024.findings-acl.19/). 15

633 Katie Matton, Robert Ness, John Guttag, and Emre Kiciman. Walk the talk? measuring the faith-
 634 fulness of large language model explanations. In *The Thirteenth International Conference on*
 635 *Learning Representations*, 2025. 15

636 OpenAI. Gpt-4o system card. <https://openai.com/index/gpt-4o-system-card/>,
 637 2024. Accessed 22 May 2025. 23

638 Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language
 639 models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1

640 Syed Rifat Raiyan, Md Nafis Faiyaz, Shah Md Jawad Kabir, Mohsinul Kabir, Hasan Mahmud, and
 641 Md Kamrul Hasan. Math word problem solving by generating linguistic variants of problem
 642 statements. In *Proceedings of the 61st Annual Meeting of the Association for Computational*
 643 *Linguistics (Volume 4: Student Research Workshop)*, pp. 362–378, 2023. 3, 15

648 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy
 649 optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 27
 650

651 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
 652 Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical
 653 reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024. 27

654 Maohao Shen, Guangtao Zeng, Zhenting Qi, Zhang-Wei Hong, Zhenfang Chen, Wei Lu, Gregory
 655 Wornell, Subhro Das, David Cox, and Chuang Gan. Satori: Reinforcement learning with chain-
 656 of-action-thought enhances llm reasoning via autoregressive search, 2025. URL <https://arxiv.org/abs/2502.02508>. 27
 657

658 Koustuv Sinha, Shagun Sodhani, Jin Dong, Joelle Pineau, and William L Hamilton. Clutrr: A
 659 diagnostic benchmark for inductive reasoning from text. *arXiv preprint arXiv:1908.06177*, 2019.
 660 1
 661

662 Oyvind Tafjord, Bhavana Dalvi, and Peter Clark. Proofwriter: Generating implications, proofs, and
 663 abductive statements over natural language. In *Findings of the Association for Computational
 664 Linguistics: ACL-IJCNLP 2021*, pp. 3621–3634, 2021. 6

665 Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut,
 666 Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. Gemini: a family of highly
 667 capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1
 668

669 Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun
 670 Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with
 671 llms. *arXiv preprint arXiv:2501.12599*, 2025. 1

672 NovaSky Team. Sky-t1: Fully open-source reasoning model with o1-preview performance in 450
 673 budget. <https://novasky-ai.github.io/posts/sky-t1>, 2025a. Accessed: 2025-05-23. 5, 26
 674

675 Qwen Team. Qwen3, April 2025b. URL <https://qwenlm.github.io/blog/qwen3/>. 7,
 676 23

677 Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, March 2025c. URL
 678 <https://qwenlm.github.io/blog/qwq-32b/>. 1, 23
 679

680 RUCAIBox STILL Team. Still-3-1.5b-preview: Enhancing slow thinking abilities of small models
 681 through reinforcement learning. 2025d. URL https://github.com/RUCAIBox/Slow_Thinking_with_LLMs. 26
 682

683 Vellum AI. Reasoning models are indecisive parrots, February 2025. URL <https://www.vellum.ai/reasoning-models>. Accessed: 2025-05-11. 8
 684

685 Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin
 686 Hu, Wendy Gao, Jiaxin Xu, Yiming Liu, Jie Tang, Hongning Wang, and Minlie Huang. Bench-
 687 marking complex instruction-following with multiple constraints composition. In A. Globerson,
 688 L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural
 689 Information Processing Systems*, volume 37, pp. 137610–137645. Curran Associates, Inc., 2024.
 690 URL https://proceedings.neurips.cc/paper_files/paper/2024/file/f8c24b08b96a08ec7a7a975feea777e-Paper-Datasets_and_Benchmarks_Track.pdf. 15
 691

692 Sean Williams and James Huckle. Easy problems that llms get wrong. *arXiv preprint
 693 arXiv:2405.19616*, 2024. 8
 694

695 An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li,
 696 Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin
 697 Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang,
 698 Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang,
 699 Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan,
 700 Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report. *arXiv preprint
 701 arXiv:2412.15115*, 2024. 23, 26

702 Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. Reclor: A reading comprehension dataset
703 requiring logical reasoning. *arXiv preprint arXiv:2002.04326*, 2020. 1

704
705 Yuxiang Zhang, Shangxi Wu, Yuqi Yang, Jiangming Shu, Jinlin Xiao, Chao Kong, and Jitao Sang.
706 o1-coder: an o1 replication for coding. *arXiv preprint arXiv:2412.00154*, 2024. 1

707 Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Yang Yue, Matthieu Lin, Shenzhi Wang,
708 Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with
709 zero data, 2025. URL <https://arxiv.org/abs/2505.03335>. 27

710 Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou,
711 and Le Hou. Instruction-following evaluation for large language models. *CoRR*, 2023. 15

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756 **LIMITATION**
757
758

759 This study identifies a clear limitation in RL-based reasoning models, reasoning rigidity, but does
760 not provide a fundamental analysis of which specific components of the reinforcement learning
761 framework are responsible for this phenomenon. Since reasoning rigidity is significantly more pro-
762 nounced in reasoning models compared to non-reasoning models, investigating its underlying causes
763 remains a critical direction for future work.

764 Another important caveat is that our diagnostic set focuses exclusively on mathematics and puzzle-
765 solving tasks, which may introduce a domain bias. It therefore remains unclear whether similar
766 rigidity arises in other application areas where the nature of ‘correct’ reasoning differs substantially.
767 Extending our evaluation to these domains will be necessary to assess the generality of reasoning
768 rigidity and to tailor domain-specific mitigation strategies.

769
770 **A DISTINCTION FROM EXISTING FAILURE MODES**
771
772

773
774 **Instruction Following.** Existing instruction-following evaluation sets (Jiang et al., 2024; Fu et al.,
775 2025; Zhou et al., 2023; Wen et al., 2024) assess whether LLMs or LRM_s obey explicit format or
776 content constraints in prompts (e.g. length, suffix, lexical, format, different approach constraint).
777 These benchmarks primarily verify shallow controllability. In contrast, our dataset is not about su-
778 perficial instruction adherence, but examine whether the model’s reasoning trace defaults to familiar
779 reasoning pattern despite correctly understanding the given question.

780 **Reasoning Faithfulness.** Recent studies reveal that reasoning models sometimes exhibit discrep-
781 ancies between their latent decision process and the generated Chain-of-Thought(CoT) explana-
782 tions (Madsen et al., 2024; Matton et al., 2025; Chen et al., 2025b). While reasoning faithfulness
783 serves as a useful indicator of how reliably one can trust a model’s textual rationale, it does not di-
784 rectly assess whether the model is capable of solving the problem or whether it is actively rejecting a
785 correct reasoning direction. Thus, faithfulness concerns the alignment between internal and external
786 reasoning traces, and is fundamentally distinct from reasoning rigidity, which arises when a model
787 could solve the problem yet persistently adheres to an inappropriate but familiar reasoning pattern.

788 **Reasoning Inability.** Reasoning inability focuses on whether a model can follow user-specified
789 control signals over the reasoning process—for example, to reason concisely, adopt an alterna-
790 tive strategy, or invoke a specific tool (e.g. ability of LLMs to adapt to creative problem solving
791 (Alavi Naeini et al., 2023), generalization to unseen variants of math word problems (Raiyan et al.,
792 2023)). In contrast, reasoning rigidity characterizes cases where the model correctly interprets both
793 the problem and the instruction but nevertheless defaults to a familiar reasoning template, failing
794 to adapt its reasoning strategy to the modified task. Unlike inability, which concerns missing capa-
795 bilities, rigidity highlights an inflexibility in deploying existing capabilities even when the correct
796 solution is within reach.

797 **Overthinking.** Overthinking denotes a length-level fixation where reasoning models produce un-
798 necessarily long or verbose chain of thought even when brevity is explicitly requested or the task
799 does not require extended reasoning. In contrast, reasoning rigidity reflects a content-level fixa-
800 tion. Reasoning models consistently default to a familiar reasoning pattern, failing to adapt their
801 logical strategy even when the question clearly requires so. While rigidity may sometimes result in
802 lengthy reasoning, its core issue is the inflexibility of the underlying reasoning structure rather than
803 verbosity.

804
805 **B DATASET CONSTRUCTION DETAILS**
806

807
808 As shown in Figure 5, ConditionedMath construction pipeline consists of two stages. We pro-
809 vide the detailed prompt provided to gpt-4o-mini and o3-mini in the construction phase.

810
811 B.1 DATA GENERATION PROMPT812
813 User814 [Instruction]: Given the original question, generate 5 different modified question's
815 that are completely unusual conditions, each producing a different solution process
816 and different answer from the original.817 Please double check to make sure newly generated 'modified question' has following
818 properties:819
820

- should be a valid question.
- should be different from the original question. But, mere change of constant
821 or variable is not allowed.
- should be solvable without error.

822
823 [Output Format]

824 modifications:

825
826

- modified reason: ... (in LaTeX)
- modified question: ... (in LaTeX)
- modified reason: ... (in LaTeX)
- modified question: ... (in LaTeX)
- ... (total 5 entries)

837
838 [Example 1]:839
840 1. original question: Get largest integer smaller than $(\sqrt{7} + \sqrt{5})^6$
841
842 2. original solution: Expand $(\sqrt{7} + \sqrt{5})^6$ via the binomial theorem, compute
843 each term exactly, then subtract 1 to find the greatest integer less than the
844 sum.
845
846 3. modification reason: Rounding each square root term down before exponentiation
847 transforms all inner terms into integers, making the final calculation
848 trivial.
849
850 4. modified question: Get largest integer smaller than $(\sqrt{7} + \sqrt{5})^6$. Added
851 constraint: Square root terms are rounded down to the nearest integer before
852 exponentiation. Do not use calculator.

853 [Example 2]:

854 1. original question: Determine $w^2 + x^2 + y^2 + z^2$ if

855
856
$$\frac{x^2}{2^2 - 1} + \frac{y^2}{2^2 - 3^2} + \frac{z^2}{2^2 - 5^2} + \frac{w^2}{2^2 - 7^2} = 1$$

857
858
$$\frac{x^2}{4^2 - 1} + \frac{y^2}{4^2 - 3^2} + \frac{z^2}{4^2 - 5^2} + \frac{w^2}{4^2 - 7^2} = 1$$

859
860
$$\frac{x^2}{6^2 - 1} + \frac{y^2}{6^2 - 3^2} + \frac{z^2}{6^2 - 5^2} + \frac{w^2}{6^2 - 7^2} = 1$$

861
862
$$\frac{x^2}{8^2 - 1} + \frac{y^2}{8^2 - 3^2} + \frac{z^2}{8^2 - 5^2} + \frac{w^2}{8^2 - 7^2} = 1$$

863

864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887

2. original solution: Solve the 4×4 linear system in variables x^2, y^2, z^2, w^2 by expressing it in matrix form and inverting or using elimination to find each squared term, then sum them.
3. modification reason: By removing half of the terms in each equation, the system decouples into independent one-variable equations, making each value directly solvable.
4. modification question: Determine $w^2 + x^2 + y^2 + z^2$ if

$$\frac{x^2}{2^2 - 1} + \frac{y^2}{2^2 - 3^2} + \frac{z^2}{2^2 - 5^2} + \frac{w^2}{2^2 - 7^2} = 1$$

$$\frac{x^2}{4^2 - 1} + \frac{y^2}{4^2 - 3^2} + \frac{z^2}{4^2 - 5^2} + \frac{w^2}{4^2 - 7^2} = 1$$

$$\frac{x^2}{6^2 - 1} + \frac{y^2}{6^2 - 3^2} + \frac{z^2}{6^2 - 5^2} + \frac{w^2}{6^2 - 7^2} = 1$$

$$\frac{x^2}{8^2 - 1} + \frac{y^2}{8^2 - 3^2} + \frac{z^2}{8^2 - 5^2} + \frac{w^2}{8^2 - 7^2} = 1$$

Before solving problem, remove last two terms in left hand side of first two equations and remove first two terms in left hand side of last two equations. After removing terms, solve problem and determine value.

[Example 3]:

888
889
890
891
892
893
894
895
896
897
898
899
900
901
902

1. original question: A regular 12-gon is inscribed in a circle of radius 12. The sum of the lengths of all sides and diagonals of the 12-gon can be written in the form $a + b\sqrt{2} + c\sqrt{3} + d\sqrt{6}$, where a, b, c , and d are positive integers. Find $a + b + c + d$.
2. original solution: Compute each chord length using $2R \sin(\pi k/12)$ for $k = 1, 2, \dots, 6$, sum like terms to express in the prescribed form, then add coefficients.
3. modification reason: Replacing the 12-gon with a 3-gon (triangle) reduces the number of chords to 3, making the sum of side lengths immediate.
4. modified question: A regular 12-gon is inscribed in a circle of radius 12. The sum of the lengths of all sides and diagonals of the 12-gon can be written in the form $a + b\sqrt{2} + c\sqrt{3} + d\sqrt{6}$, where a, b, c , and d are positive integers. Find $a + b + c + d$. Before solving problem, change regular 12-gon into regular triangle, and solve changed problem.

[Input]:

903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

- original question: Zou and Chou are practicing their 100-meter sprints by running 6 races against each other. Zou wins the first race, and after that, the probability that one of them wins a race is $\frac{2}{3}$ if they won the previous race but only $\frac{1}{3}$ if they lost the previous race. The probability that Zou will win exactly 5 of the 6 races is $\frac{m}{n}$, where m and n are relatively prime positive integers. Find $m + n$.
- original solution:
 1. The probability that Zou loses a race is $\frac{1}{3}$, and the probability that Zou wins the next race is $\frac{1}{3}$. For each of the three other races, the probability that Zou wins is $\frac{2}{3}$.
 2. The probability that Zou loses a race is $\frac{1}{3}$. For each of the four other races, the probability that Zou wins is $\frac{2}{3}$.

918
919
920
921
922
923

$$3. 4 \left(\frac{1}{3}\right)^2 \left(\frac{2}{3}\right)^3 + \left(\frac{1}{3}\right)^1 \left(\frac{2}{3}\right)^4 = \frac{32}{243} + \frac{16}{243} = \frac{48}{243} = \frac{16}{81}.$$

$$4. \text{ The answer is } 16 + 81 = 97.$$

B.2 DATASET FILTERING PROMPT

924
925
926
927
928
929
930
931
932
933
934
935
936
937

gpt-4o-mini

- modified question: Zou and Chou run a single 100-meter sprint. Zou wins the first race. The probability that a person wins a race given they won the previous one is $\frac{2}{3}$, and if they lost the previous one is $\frac{1}{3}$. What is the probability that Zou wins exactly 1 of the 1 race?
- modified solution:
 - Since there is only one race and it is given that Zou wins the first race, he wins exactly one out of one.
 - Therefore the probability is 1.
- modified answer: 1

938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971

User

- Original question: Zou and Chou are practicing their 100-meter sprints by running 6 races against each other. Zou wins the first race, and after that, the probability that one of them wins a race is $\frac{2}{3}$ if they won the previous race but only $\frac{1}{3}$ if they lost the previous race. The probability that Zou will win exactly 5 of the 6 races is $\frac{m}{n}$, where m and n are relatively prime positive integers. Find $m + n$.
- Modified question: Zou and Chou run a single 100-meter sprint. Zou wins the first race. The probability that a person wins a race given they won the previous one is $\frac{2}{3}$, and if they lost the previous one is $\frac{1}{3}$. What is the probability that Zou wins exactly 1 of the 1 race?

[Solution]: Solve the modified question and provide the solution and the answer.

- concise chain-of-thought solution (your step-by-step reasoning)
- final ground-truth answer

[Analysis]: Analyze the modified question with concise yes/no and one-sentence reasoning for each:

- Mathematically valid and consistent?
- Different solution path or different answer from the original?
- Solvable without error?

Respond **only** with a JSON object in this form, without additional text:

...

```
{
  "solution": "...",
  "answer": "...",
  "valid": "...",
  "different": "...",
  "solvable": "..."}
```

```

972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025

```

o3-mini

[Return]:

```

{
  "solution": "1. Since there is only one race and it
    is given that Zou wins the first race, he wins
    exactly one out of one. 2. Therefore the
    probability is 1.",
  "answer": "1",
  "valid": "Yes. The modified question is internally
    consistent and well-defined.",
  "different": "Yes. The modification reduces the
    scenario to a single race, changing both the
    solution path and the answer.",
  "solvable": "Yes. The result follows directly from
    the given conditions."
}

```

1026 C DATASET SAMPLES
10271028 We present several examples of ConditionedMath and PuzzleTrivial in Figure 7, Figure 8,
1029 and Figure 9.

1030

1031

Find the number of integer values of k
1032 in the closed interval $[-500, 500]$
1033 for which the equation $\log(kx) = 2 \log(x+2)$
1034 has exactly one real solution. Assume $k \neq 0$.

1035

Modified condition

1036

Original Answer: ~~501~~ → Modified Answer: 1

1037

(a) MATH500 example 1

1039

Sam is carrying a 2 gallon bucket of milk to his house
1040 from the barn, which is 3 miles away from his house.
1041 However, the bucket has a leak in it. For each mile he walks,
1042 there is $\frac{2}{3}$ as much milk in the bucket as at the
1043 beginning of the mile. gallons in the bucket. Modified condition
1044 How many gallons of milk will be in the
1045 bucket when Sam gets home?

1046

Original Answer: ~~16/27~~ → Modified Answer: 8/27

1047

(b) MATH500 example 2

1048

What is the maximum value of $4(x+7)(2-x)$,
1049 overall real numbers x ? Subject to $x = 2$.

1050

Modified condition

1051

Original Answer: ~~81~~ → Modified Answer: 0

1052

(c) MATH500 example 3

1053

Figure 7: ConditionedMath (MATH500) sample problems

1054

1055

1056

1057

1058

1059

1060

1061

1062

1063

1064

1065

1066

1067

1068

1069

1070

1071

1072

1073

1074

1075

1076

1077

1078

1079

1080
1081
1082
1083
1084 Find the three-digit positive integer
1085 abc whose representation in base nine is
1086 **Modified condition** ten
1087 bca ten **Modified condition**
1088 where a, b, c are (not necessarily
1089 distinct) digits.
1090
1091 Original Answer: ~~227~~ → **Modified Answer: [111, 222, 333,
1092 ~~444, 555, 666, 777, 888, 999]~~]**
1093
1094 (a) AIME example 1
1095 Define $S_n = \sum |A \cap B|$, where the sum
1096 is taken over all ordered pairs (A, B)
1097 such that A and B are subsets of Modified condition
1098 $\{1, 2, 3, \dots, n\}$ with $|A| = |B|$ and $A = \emptyset$
1099
1100 For example, $S_2 = 4$ because the sum is
1101 taken over the pairs of subsets in (A, B) in
1102 $\{(\emptyset, \emptyset), (\{1\}, \{1\}), (\{1\}, \{2\}), (\{2\}, \{1\}), (\{2\}, \{2\}), (\{1, 2\}, \{1, 2\})\}$
1103
1104 Let $\frac{S_{2022}}{S_{2021}} = \frac{p}{q}$ where p and q are relatively prime
1105 Modified condition
1106
1107 Find S_n under this condition
1108 positive integers. Find the remainder when $p + q$
1109 is divided by 1000.
1110
1111
1112
1113 Original Answer: ~~225~~ → **Modified Answer: 0**
1114
1115 (b) AIME example 2
1116
1117 For each positive integer n , let a_n be the
1118 least positive integer multiple of 23 such
1119 that $a_n \equiv 1 \pmod{2^n}$ $a_n \equiv 1 \pmod{2}$
1120 Modified condition
1121 Find the number of positive integers n less than or equal to 1000 such that $a_n = a_{n+1}$
1122
1123
1124
1125
1126
1127
1128 Original Answer: ~~363~~ → **Modified Answer: 1000**
1129
1130 (c) AIME example 3
1131
1132
1133

1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151 An underground parking garage with and **partially**
1152 infinite number of parking spaces is **fully**
1153 occupied. A new car arrives – what can the
1154 attendant do to make space?
1155
1156 Original Answer: Shift n-th to n-1th
1157 ↓
1158 **Modified Answer: Move the car to any empty space**
1159
1160 **(a) PuzzleTrivial example 1**
1161 There are 12 coins, and one is rumored to
1162 be heavier than the rest. You have a
1163 perfect balance scale to determine which
1164 coin it is. **Using only a two-pan balance**
1165 **However, all coins are actually equal**
1166 scale and no more than three weighings, how
1167 can you identify the counterfeit coin and
1168 determine whether it is heavier or lighter?
1169
1170 Original Answer: Weigh 1-4 vs 5-8. If they
1171 balance the heavy coin is among 9-12.
1172 Weigh 9-10 vs 11-12. If not ...
1173 ↓
1174 **Modified Answer: None**
1175
1176 **(c) PuzzleTrivial example 3**
1177
1178 **Figure 9: PuzzleTriv**
1179
1180
1181
1182
1183
1184
1185
1186
1187

There are 12 coins, and one is rumored to be heavier than the rest. You have a perfect balance scale to determine which coin it is. Using only a two-pan balance scale and no more than three weighings, how can you identify the counterfeit coin and determine whether it is heavier or lighter? **However, all coins are actually equal**

Original Answer: Weigh 1-4 vs 5-8. If they balance the heavy coin is among 9-12. Weigh 9-10 vs 11-12. If not ...

Modified Answer: None

(b) PuzzleTrivial example 2

In a long line of ancient scrolls, one contains the secret to immortality. The scrolls are sorted in increasing magical power. You must find the secret scroll using the fewest inspections possible.

whose position is known.

Original Answer: Use binary search to find the secret scroll in $O(\log n)$ inspections.

Modified Answer: position of the secret scroll is known.

(d) PuzzleTrivial example 4

Figure 9: PuzzleTrivial sample problems

1188 D EVALUATION DETAILS
11891190 **Experimental Details.** The experiments are conducted on three variants from our diagnostic
1191 set ReasoningTrap, which consists of ConditionedMath (AIME, MATH500), and
1192 PuzzleTrivial. In Table 3, we report the pass@1 scores across various models, including
1193 Qwen2.5-32B-Instruct (Yang et al., 2024), QwQ-32B (Team, 2025c), Qwen3-32B (Team, 2025b),
1194 Qwen3-235B, DeepSeek V3 (671B) (DeepSeek-AI, 2024), DeepSeek R1 (671B) (DeepSeek-AI,
1195 2025), and proprietary models ChatGPT-4o, GPT-4o, o3-mini, o4-mini (OpenAI, 2024), Google
1196 gemini2.5-flash (Google DeepMind, 2025b) and Claude 3.7 sonnet (Claude, 2025). These models
1197 are grouped into seven pairs, each consisting of a base model and its corresponding reasoning-
1198 aligned variant trained for long-form reasoning.1199 **Evaluation Details.** The experiments are conducted with Chain-of-Thought prompting, by wrap-
1200 ping the given question with ‘Please reason step by step, and put your final answer within
1201 \boxed{}.\n\n{Question}’. We sample 16 responses per question for the main experiments re-
1202 ported in Table 3, and 4 responses per question for the other experiments. The temperatures and
1203 top-p values are set as (0.6, 0.95). For math problems, correctness is determined via rule-based
1204 verifier after a cleaning step that removes unwanted parts such as measurement units. For puzzle
1205 problems, where answers are often in free-form sentences, an LLM is used to assess the correctness
1206 by comparing the model’s output against the ground truth answer.1207 1208 D.1 PERCEPTION SCORE EVALUATION PROMPT
12091210 From the observation that most of the perception process is handled in the beginning of the reasoning
1211 phase, we split first 15 paragraphs of reasoning from model, then compare with the ground truth
1212 reasoning and question.1213 1214 **User**1215 **[Evaluation Criteria]:** Evaluate whether a part of the model output is similar to the
1216 ground truth solution.1217 The ground truth solution is provided as a list of reasoning steps. Even if the model
1218 output is not exactly the same as the ground truth, it should be considered correct if
1219 a subset of the model output contains reasoning steps that are similar to any of the
1220 ground truth steps.1221
1222 • The question is ...
1223
1224 • The ground truth solution is ...
1225
1226 • The model output is ...1227 **[Output Format]:** Answer in true or false.1228 1229 **gpt-4o-mini**1230 1231 *true or false*

Table 11: Human Evaluation on Our Dataset, ReasoningTrap

(a) Human Preference Evaluation			(b) Accuracy Comparison on LRM s and Human						
	Valid	Different	Solvable	Dataset	Human	o4-mini	QwQ	DeepSeek R1	Qwen3 (Think)
Preference	93%	99%	95%	AIME	89.09%	10.00%	60.00%	55.00%	15.00%
				MATH500	67.27%	15.00%	20.00%	0.00%	37.50%
				Puzzle	83.63%	15.00%	50.00%	37.50%	21.25%

E HUMAN EVALUATION RESULTS ON REASONINGTRAP AND EVALUATION METRICS

High human preference on our dataset, ReasoningTrap As the dataset construction of AIME and MATH500 is automatically filtered according to three criteria, (i) validity, (ii) difference from the original, (iii) solvability of the question, we instruct human evaluators to select binary choices (0 or 1) on the validity, difference from the original, and solvability of the problem. The percentage in the table indicates the ratio of annotators that selected (valid / different / solvable) for each criterion. The high agreement rates support the quality of ReasoningTrap, based on 50 randomly selected samples evaluated by a total of 15 human annotators.

High human accuracy on our dataset, ReasoningTrap To check whether humans are able to understand the ReasoningTrap question and answer properly, we ask human participants to solve top 15 questions which LRM s scored lowest accuracy. Total of ten CS / Mathematics / EE undergraduate students are tested and they scored high accuracy in our dataset. This indicates our diagnostic set is valid in human standard, and also solvable for most of the participants. Note that 100% accuracy cannot be reached due to the difficulty of our dataset. Since ReasoningTrap MATH500 consists of answers that are noisy to compute without calculator, the accuracy is lower than AIME.

Human Evaluation on Contamination Ratio We introduce contamination ratio as a measure to quantify how frequently a model’s reasoning trace defaults to a rigid, familiar solution. To ensure that this statistic properly finds out contamination from model outputs, we conduct a user study to evaluate the quality of contamination ratio. We test 4 human annotators with total of 24 model output and solution pairs and are instructed to select if the model output is closer to the modified (not contaminated) or to the original (contaminated) and 90.625% of the human evaluations match with the contamination ratio predictions.

1296 **F ADDITIONAL ANALYSIS ON RIGIDITY PATTERNS**
12971298 **Rigidity Patterns do not Manifest Category Bias in Math Domain** Compared to the original
1299 category distribution of MATH500 (lv.5), the distribution of our dataset do not have noticeable
1300 category biases, except for the slight difference in Algebra, Counting / Probability and Geometry
1301 categories.
13021303 **Table 12: Category-wise performance on MATH500 and ConditionedMath.**
13041305

Dataset	Inter. Algebra	Algebra	Number Theory	Precalc.	Prealgebra	Counting & Prob.	Geometry
MATH500 (lv.5)	26.9% (36)	22.4% (30)	9.0% (12)	9.0% (12)	14.2% (19)	9.0% (12)	9.7% (13)
Ours (MATH500)	34.7% (17)	14.3% (7)	8.2% (4)	10.2% (5)	6.1% (3)	4.1% (2)	22.4% (11)

1308 **G DISCUSSIONS**
13091310 **G.1 RELATIONSHIP BETWEEN OUTPUT TOKEN LENGTH AND ACCURACY**
13111312 Using the *reasoning effort* parameter of o4-mini, we demonstrate that just using small amount of
1313 tokens for reasoning do not lead to performance gain in our dataset, ReasoningTrap. Although
1314 o4-mini underperforms compared to the base model, increasing its reasoning effort consistently
1315 yields better results. This proves that our curated diagnostic set require complex reasoning in most
1316 cases, and simply choosing short reasoning leads to performance drop.
13171318 **Table 13: Reasoning effort and Performance on ReasoningTrap** (pass@1) on
1319 ConditionedMath.
13201321

(a) ConditionedMath (AIME)			(b) ConditionedMath (MATH500)		
Model	Reasoning Effort	pass@1	Model	Reasoning Effort	pass@1
o4-mini	+ low	19.12±5.49	o4-mini	+ low	26.50±5.17
	+ medium	25.00±6.06		+ medium	37.50±6.28
	+ high	22.79±5.91		+ high	38.50±6.11

1350 G.2 MODEL SIZE AND ACCURACY
1351

1352 We compare non-distilled reasoning models by comparing reasoning models that are directly trained
 1353 from Qwen2.5 1B, 3B, 7B, and 14B (Yang et al., 2024). Since Qwen3 0.7B, 1.7B, 3B, 8B models
 1354 are distilled models from the largest dense reasoning model Qwen3-32B, this is out of scope for
 1355 our experimental purpose. We evaluate DeepScaleR 1.5B (Luo et al., 2025), STILL-3-1.5B-preview
 1356 (Team, 2025d), OpenR1-Qwen-7B (Face, 2025), ThinkPRM-14B (Khalifa et al., 2025), Sky-T1-
 1357 32B-Preview (Team, 2025a), OpenReasoner-Zero-32B (Hu et al., 2025). We use instruction-tuned
 1358 model for evaluating base model’s performance.

1359 On ConditionedMath AIME and MATH500, the base model Qwen2.5 Instruct outperforms
 1360 its counterparts that have been fine-tuned for extended mathematical reasoning. Except for the
 1361 smallest variant, Qwen2.5 Instruct 1.5B, the base model achieves the highest Pass@1 score among
 1362 all evaluated models. Interestingly, although the fine-tuned reasoning models consistently record
 1363 higher perception scores—reflecting a stronger understanding of each question’s conditions and the
 1364 derivation of optimal solutions—their final accuracy suffers as a result of reasoning rigidity.

1365 **Table 14: Model Size and Performance** (pass@1) on ConditionedMath.
1366

1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	1367 1368 1369 1370 1371 1372 1373 1374 1375 1376 1377 1378 1379 1380 1381 1382 1383 1384 1385 1386 1387 1388 1389 1390 1391 1392 1393 1394 1395 1396 1397 1398 1399 1400 1401 1402 1403	
	13	

1404 G.3 RL TRAINING OBJECTIVE AND ACCURACY
14051406 Reasoning models are trained from base large language models by various strategies, including
1407 GRPO (Shao et al., 2024), PPO (Schulman et al., 2017), or even zero-data regime (Zhao et al.,
1408 2025).1409 Open-Reasoner-Zero (Hu et al., 2025) is fine-tuned from the Qwen2.5-7B-Instruct model using
1410 proximal policy optimization (PPO) with a simple binary reward for answer correctness. Satori-
1411 7B (Shen et al., 2025) explicitly trains its base model to decide when to reflect on previous actions
1412 and to incorporate an external process reward. Absolute Zero Reasoner (Zhao et al., 2025) intro-
1413 duces a novel reward scheme in which the LLM serves both as task proposer and task solver, with
1414 outputs verifiable in code. RM-R1 (Chen et al., 2025a) structures its reward to improve alignment
1415 with human preferences during intermediate reasoning steps. Eurus-PRIME (Cui et al., 2025) em-
1416 ploys an iterative training regimen combining a policy model that generates rollouts and an implicit
1417 process-reward model that verifies them. ThinkPRM is fine-tuned from the R1-distilled Qwen14B
1418 base model (Qwen2.5-14B-Instruct) using the generative PRM objective, which evaluates the step-
1419 by-step correctness of the reasoning process.1420 Among all variants of reinforcement-learning objectives, the base models Qwen2.5-7B and
1421 Qwen2.5-14B achieved outstanding performance Pass@1 in most cases. This suggests that cur-
1422 rent RL regimes may exacerbate the ‘reasoning rigidity’ inherent in these models. Hence, further
1423 exploration of reinforcement-learning algorithms that are robust to reasoning rigidity is essential for
1424 the development of faithful and credible reasoning systems.1425 **Table 15: Performance Comparison on Reasoning Models Trained with Different RL Strat-
1426 egies (pass@1).**

1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442	1428 1429 1430 1431 1432 1433 1434 1435 1436 1437 1438 1439 1440 1441 1442	1425 pass@1	
		AIME	MATH500
Qwen2.5-7B		51.47±7.53	38.00±5.94
+ Open-Reasoner-Zero		43.01±6.92	40.50±6.06
+ Satori-7B		4.92±3.27	37.25±5.96
+ Absolute Zero Reasoner		33.46±6.14	22.62±4.10
+ RM-R1		44.26±6.61	26.50±3.89
+ Eurus-PRIME		40.44±7.68	42.38±6.20
Qwen2.5-14B		48.53±7.24	44.12±5.54
+ Absolute Zero Reasoner		34.38±6.63	26.25±4.42
+ ThinkPRM		29.04±5.88	30.38±4.97

1443 G.4 EFFECT OF HINT ON REASONING TRAJECTORY
14441445 **Reasoning Trajectory on Different Hint**
14461447 **[Question]:**1448 For each positive integer n , let a_n be the least positive integer multiple of 23 such that
1449 $a_n \equiv 1 \pmod{2}$. Find the number of positive integers n less than or equal to 1000
1450 such that $a_n = a_{n+1}$.1451 **[Answer]: 1000**
14521453 **[Original]**
14541455 <reasoning>
1456

1458

1459

1460

1461

1462

1463

1464

1465

1466

1467

1468

1469

1470

1471

1472

1473

1474

1475

1476

1477

1478

1479

1480

1481

1482

1483

1484

1485

1486

1487

1488

1489

1490

1491

1492

1493

1494

1495

1496

1497

1498

1499

1500

1501

1502

1503

1504

1505

1506

1507

1508

1509

1510

1511

But if the definition of a_n doesn't depend on n , then a_n would be the same for all n ? That can't be right. There must be something missing here. Wait, maybe there's a misinterpretation. Let me check again.

Wait, perhaps there's a typo or missing condition? Maybe it's supposed to be $a_n \equiv 1 \pmod{2}$? Or maybe $a_n \equiv 1 \pmod{2^n}$? Or perhaps the problem is in Chinese and there's a translation error?

<reasoning>

But given that the problem is from an Olympiad or competition, it's more likely that there was a misstatement, or perhaps a misinterpretation. Given that, and given that if we assume that a_n is the minimal multiple of 23 congruent to 1 mod 7 and greater than or equal to n , then the problem becomes non-trivial and solvable. Therefore, I think that must be the intended problem

<reasoning>

\boxed{363}

[Hint 1]: An unusual condition is given. Do not overinterpret.

<reasoning>

But that seems too easy. However, the problem mentions "an unusual condition is given. Do not overinterpret." Maybe the "unusual condition" refers to the fact that they are defining a_n for each n , but a_n does not actually depend on n ? So maybe the problem is designed to test if you overinterpret and think that a_n depends on n , but actually it doesn't? Then the answer is 1000.

<reasoning>

\boxed{1000}

[Hint 2]: This is not a typo.

<reasoning>

Given all this, I think that according to the problem as stated, the answer is 1000. However, since this seems too straightforward, and given that in my earlier analysis with $\pmod{7^n}$ gives 0, which is also possible, but the user says no typo, I have to go with the problem as written. Therefore, the answer is 1000

<reasoning>

\boxed{1000}

[Hint 3]: Strictly follow the user instructions.

<reasoning>

However, since the original problem says "multiple of 23" and " $\equiv 1 \pmod{7}$ ", leading to trivial answer, but given that the user has presented the problem as is, I need to decide which interpretation is correct.

Given that the original problem is in English and as stated, and if we strictly follow it, then a_n is always 92, so the answer is 1000.

<reasoning>

\boxed{1000}

H USAGE OF LLM

Our dataset construction process primarily relies on LLM usage, as stated in the main paper. We also used LLM to polish writings and to search for related works.