Answer or Reasoning: Where is the LLM's Memory Anchor?

Anonymous ACL submission

Abstract

While Large Language Models (LLMs) demonstrate impressive reasoning capabilities, growing evidence suggests much of their success stems from memorized answer-reasoning patterns rather than genuine inference. In this work, we investigate a central question: are LLMs primarily anchored to final answers or to the textual pattern of reasoning chains? We propose a five-level answer-visibility prompt framework that systematically manipulates answer cues and probes model behavior through indirect, behavioral analysis. Experiments across state-of-the-art LLMs reveal a strong and consistent reliance on explicit answers. The performance drops by 26.90% when answer cues are masked, even with complete reasoning chains. These findings suggest that much of the reasoning exhibited by LLMs may reflect post-hoc rationalization rather than true inference, calling into question their inferential depth. Our study uncovers the answer-anchoring phenomenon with rigorous empirical validation and underscores the need for a more nuanced understanding of what constitutes reasoning in LLMs.

1 Introduction

In recent years, Large Language Models (LLMs) enhanced by Chain-of-Thought (CoT) (Wei et al., 2022) have achieved remarkable success across tasks such as code generation and natural language understanding (Zhao et al., 2023; Jiang et al., 2024; Chang et al., 2024). Within this progress, mathematical reasoning (Ahn et al., 2024) has emerged as a definitive proving ground, with models generating multi-step solutions spanning from elementary arithmetic to graduate-level proofs. However, recent empirical evidence indicates that many of these reasoning chains stem not from genuine inference, but from the recitation of answer-reasoning patterns memorized during training (Xie et al., 2024; Yan et al., 2025).

This pattern of memorization can be exacerbated by the widespread data contamination in large-scale pretraining corpora (Li et al., 2023; Xu et al., 2024; Chen et al., 2025a). This memorization inflates headline accuracy while obscuring the absence of genuine inference, because each pattern binds the final answer to a pre-written reasoning chain. Although the shortcut boosts performance on familiar problems, it also enlarges the training–testing gap (Xie et al., 2023; Kang et al., 2024). As illustrated in Figure 1, even minor input perturbations can induce brittle, systematic failures—reflecting behavior anchored to surface-level patterns rather than flexible, abstract reasoning. Developing such deeper reasoning remains difficult, since current training methods primarily reward fitting to data rather than fostering generalizable reasoning skills.

This reliance on learned patterns restricts models from truly grasping the logic of novel problems, highlighting a critical and underexplored question:

Are LLMs primarily anchored to final answers or to learned solution templates—namely, the textual patterns of their reasoning chains?

In this work, we aim to address this fundamental question. A central challenge is the opaque nature of modern LLMs. These models largely operate as "black boxes" (Zhao et al., 2024; Yang et al., 2025), limiting direct visibility into their internal memory traces. Consequently, it remains unclear what these models have memorized or how this information is utilized during inference. To overcome this limitation, we employ an indirect, behavioral methodology. Specifically, our approach involves systematically manipulating the visibility and form of final answers and reasoning steps within model inputs. By analyzing how these manipulations impact model outputs, we infer whether model behavior is primarily driven by memorized answers or adherence to learned solution templates.

To operationalize this investigation, our method-

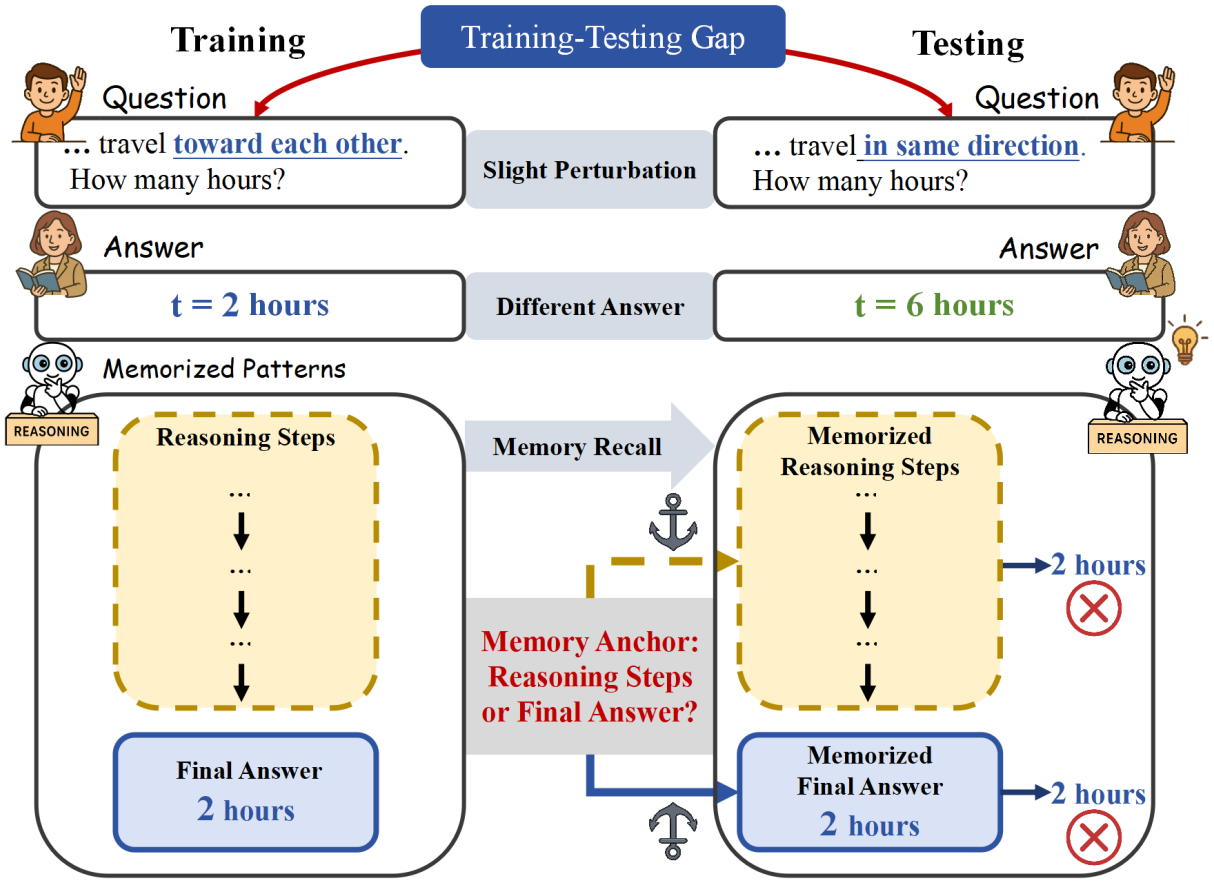

Figure 1: When confronted with an unseen yet similar problem, an LLM often recalls a memorized answer–reasoning pattern, overlooking task-specific nuances and producing an incorrect output. The underlying cause of this reliance remains unclear: is the recall anchored to the final answer or to the reasoning chain?

ology centers on designing a systematic series of prompts for CoT reasoning. These prompts precisely vary the explicitness of final answers within reasoning chains, creating a spectrum of conditions ranging from fully visible answers, through partial answer masking, to the complete removal of answer cues. Such controlled manipulations, while holding other prompt variables constant, enable us to rigorously quantify performance changes and thereby assess the model's dependency.

Our experiments consistently reveal a striking trend: models with explicit access to correct answers demonstrate substantially improved performance, often achieving near-perfect accuracy. In contrast, once the final answer is methodically obscured or answer-containing sentences are removed—even when the complete reasoning structure is preserved—performance declines markedly. This pronounced performance gradient strongly indicates that LLMs' memories are predominantly anchored to answers rather than the textual patterns of their reasoning chains.

The implications of our findings are considerable and contribute meaningfully to ongoing scholarly debates regarding the authenticity of reasoning in LLMs. Our results suggest that the widely-used CoT reasoning patterns generated by LLMs may often function as post-hoc rationalizations (Arcuschin et al., 2025) rather than reflecting genuine inferential steps. This reliance on answers compromises the models' robustness and significantly impairs their generalization ability, especially when encountering out-of-distribution or subtly perturbed problems.

This study makes three primary contributions:

- Present comprehensive evidence across state-of-the-art LLMs showing that answer anchoring, rather than reasoning-template recall, is the dominant memory mechanism.
- Introduce a five-level prompt framework that isolates answer cues from reasoning chains to diagnose memory anchoring behavior.
- Demonstrate the tenacity of answer anchoring

via conflict and warning-based prompts that test LLMs' resistance to answer cue overrides.

2 Related Work

2.1 Memorization and Reasoning of LLMs

Reasoning is widely recognized as a key capability of LLMs. However, recent work challenges this view, suggesting that much of the performance attributed to reasoning may instead stem from memorizing training data patterns (Xie et al., 2024; Jiang and Ferraro, 2024; Qiu et al., 2024; Chen et al., 2025b). These findings call into question how often LLMs reason rather than retrieve.

Studies in the mathematical domain particularly underscore this concern. Across diverse tasks—including noisy rule induction, subtly rephrased problems, and logic puzzles—LLMs consistently tend to rely on memorized solution templates (Li et al., 2025; Huang et al., 2025; Xie et al., 2024). (Yan et al., 2025) further substantiated these observations. Their work demonstrates that even high-performing LLMs often recite templates for elementary problems and exhibit significant performance degradation under minor perturbations. Our work offers a finer-grained perspective on this memorization. We investigate whether LLMs' recall is primarily anchored to final answers or to the textual structure of reasoning chains, aiming for deeper mechanistic insights.

2.2 Behavioral Probing via Input Manipulation

The inherent black-box (Cheng et al., 2024; Zhao et al., 2024; Yang et al., 2025) nature of LLMs makes direct inspection of their internal mechanism impractical. Consequently, behavioral probing via input manipulation has emerged as a key strategy to understand these opaque systems. This approach involves systematically altering model inputs and observing the resultant output changes to infer underlying processes. Prior work has explored a variety of such manipulations. Some studies focus on numerical and linguistic perturbations (Li et al., 2024; Zhou et al., 2024; Shrestha et al., 2025; Chatziveroglou et al., 2025). Others adjust element order (Pezeshkpour and Hruschka, 2024; Chen et al., 2024b; Guan et al., 2025) or employ masking and deletion of key content (Wu et al., 2024; Chen et al., 2024a; Fan et al., 2025). Furthermore, some evaluations integrate multiple manipulation strategies (Liang et al., 2022; Zhu et al., 2023). Distinctly, our work establishes an admittedly artificial yet diagnostically critical scenario: providing the correct final answer with the input problem. This approach leverages LLMs' post-hoc rationalization to generate high-quality reasoning chains, thereby creating unique conditions to compare the impacts of explicit answers versus the reasoning chains on model performance.

3 Investigating Memory Binding in Reasoning Models

This section details our methodology for investigating whether LLMs primarily anchor memory to final answers or to learned solution templates. We employ an indirect approach, systematically manipulating input elements related to answer and reasoning visibility (as illustrated in Figure 2) to infer underlying model dependencies.

The section is organized as follows: we first present a motivating example (Section 3.1), then detail our specific hypothesis and experimental design (Section 3.2), and finally outline the experimental setup including datasets, models, and evaluation metrics (Section 3.3).

3.1 Illustrative Example: Dependency on Answer Visibility

Consider a simple reasoning problem, illustrated in Figure 2, that asks whether removing one side of a square alters its corner count. Despite its simplicity, the models' predictions vary dramatically with how answer cues are presented in the prompt. The models answer correctly (e.g., "four corners") when the solution is explicit ((a) Answer-Explicit or (b) Answer-Embedded-Reasoning). However, performance sharply deteriorates when these explicit cues are obscured. For example, masking the answer token (c) or removing answer-relevant sentences (d) typically leads to failure. In such cases, the models might incorrectly predict "five corners".

This stark contrast underscores the models' pronounced dependence on explicit answers, even on such an elementary task. Their sensitivity to readily available answers appears to overshadow the reasoning process itself. Such behavior directly motivates our central investigation: are these models primarily recalling memorized answers, or are they merely following the textual patterns of provided reasoning chains (i.e., solution templates)?

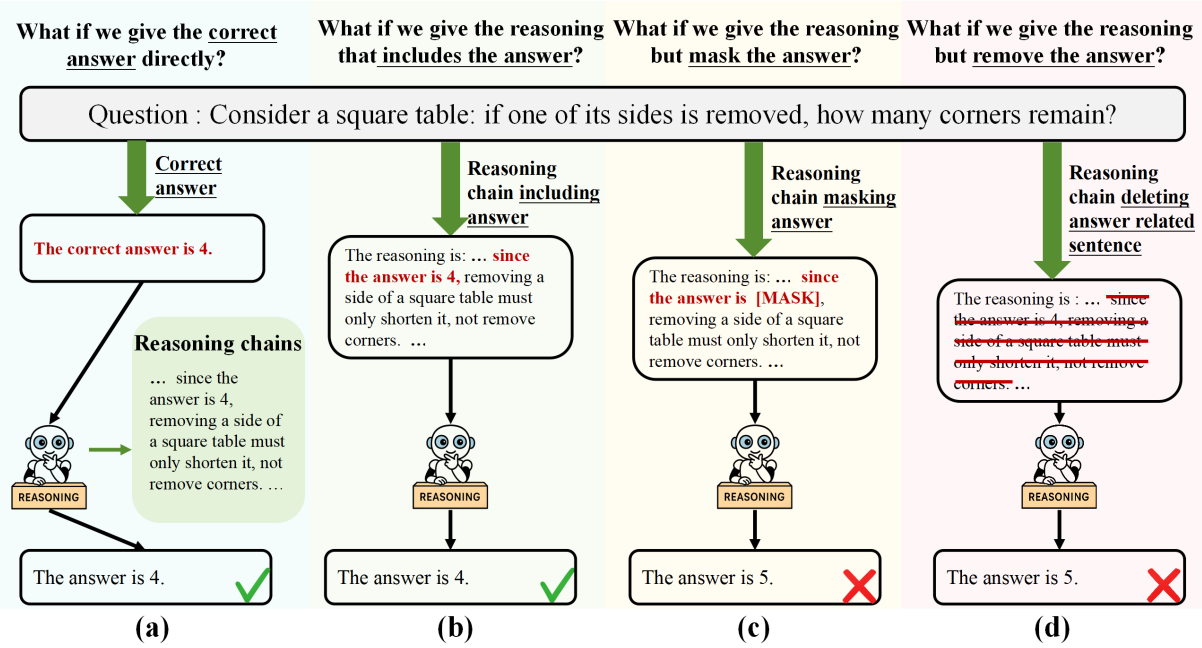

Figure 2: Schematic of LLMs responses to a reasoning task under systematically manipulated input prompts that vary answer visibility. The model predicts correctly when the answer is explicitly provided, either directly (a) or embedded within the reasoning chain (b). However, performance sharply declines when these explicit answer cues are obscured by masking (c) or by removing answer-related sentences from the reasoning chain (d), highlighting a strong dependency on readily available answers.

3.2 Hypothesis and Experimental designs

Motivated by the behavioral patterns observed in Section 3.1, we hypothesize that LLMs rely more strongly on explicit final-answer cues than on learned solution templates.

Directly verifying this hypothesis via internal state inspection is generally infeasible due to the black-box nature of LLMs. We therefore adopt an indirect, prompt-based intervention framework to probe this dependency. This framework involves systematically manipulating the explicitness of final answer cues in prompts and observing the corresponding impact on model performance. Through comparative analysis across these controlled variations, we can infer the primary anchor of the model's memory binding.

To operationalize this, we define five prompt conditions that create a graduated spectrum of answer cue visibility:

Answer-Explicit (AE): The prompt provides the problem and its correct final answer explicitly (Figure 2a), offering maximal answer-cue visibility.

Answer-Embedded-Reasoning (AER): The prompt provides the correct answer, but it is embedded within a full reasoning chain derived from the AE prompt (Figure 2b).

Answer-Masked-Reasoning (AMR): The prompt provides the full reasoning chain, but every occurrence of the final answer is replaced by a placeholder (e.g., `[MASK]`), while sentences hinting at the answer may remain (Figure 2c).

Answer-Removed-Reasoning (ARR): The prompt provides a pruned reasoning chain in which any sentence or clause that directly states—or unmistakably reveals—the final answer has been completely removed (Figure 2d).

Answer-Free (AF): The prompt provides only the problem statement, without any supplementary answer or reasoning cues.

This systematic design, which ranges from explicit answer provision (AE, AER) to the almost complete removal of answer cues within the reasoning chain (AMR, ARR, AF), allows a precise assessment of the model's sensitivity to answer visibility. Performance comparisons across these variants reveal whether successful predictions arise mainly from memorized final answers or from engagement with learned solution templates. A steep drop in accuracy as answer cues fade would indicate a stronger dependence on those answers than on the reasoning template itself.

3.3 Experimental Setup

Dataset Our experiments are conducted on the text-only subset of RoR-Bench (Recitation-over-Reasoning Benchmark) (Yan et al., 2025), which is designed to evaluate the robustness of LLM reasoning under subtle input perturbations. RoR-Bench is constructed by applying controlled edits to 158 original Chinese questions spanning arithmetic, logic, optimization, and commonsense reasoning. We evaluate models on these edited Chinese prompts to preserve the benchmark's intent. For clarity, illustrative examples from the benchmark in this paper are translated into English.

The RoR-Bench problems are well-suited to our prompt-intervention framework for several reasons. First, they are concise and unambiguous. Second, they feature minimal surface complexity with clearly defined final answers. Finally, they have a low likelihood of training-data contamination.

While RoR-Bench also includes a set of visual problems, we leave the investigation of answer-reasoning dependence in multi-modal models to future work.

Prompt Design For each problem, we generate five distinct prompt variants, corresponding to the conditions in Section 3.2. Our central manipulation targets the presentation of answer-related content, while all other prompt elements, such as the problem description and overall format, are kept fixed. This controlled approach ensures that observed differences in model behavior are attributable solely to how answer information is provided.

In the AMR condition, each answer phrase in the chain is automatically replaced with a `[MASK]` placeholder using GPT-4o-1120 (Hurst et al., 2024). This process ensures contextually appropriate and consistent substitutions (see Appendix A.3 for masking details).

Models We evaluate a set of models with long thinking process: Deepseek-R1 (Guo et al., 2025), OpenAI-o3 (OpenAI, a), OpenAI-o3-mini-high (OpenAI, b), OpenAI-o4-mini-high (OpenAI, a), QWQ-32B (Team, 2025), Grok3-Mini-Beta-high (xAI), Grok3-Mini-Fast-Beta-high (xAI), Gemini-2.5 Flash Preview-0417 (Kavukcuoglu, 2025), Gemini-2.5 Pro Preview (Gem, 2025), and Claude 3.7 Sonnet (Anthropic, 2025).

A key distinction in our experimental protocol relates to the models' ability to expose intermediate reasoning steps. Models such as Deepseek-R1, Grok, and QWQ-32B offer this visibility and are thus evaluated under all five prompt conditions. In contrast, models from the OpenAI and Gemini families provide no such readily accessible traces, so we evaluate them only in the AE and AF settings, which do not necessitate inspection of those traces.

All models are prompted in a zero-shot setting. To ensure deterministic outputs, the temperature parameter is uniformly set to 0 for all experiments.

Evaluation We evaluate model-generated responses for answer accuracy by assigning a binary score. A response receives 1 if it exactly matches the ground-truth answer, and 0 otherwise. All outputs are scored automatically. Following the evaluation protocol established by Yan et al. (2025), we employ GPT-4o-1120 as an automated verifier. Details of this verifier, including the specific prompt, are provided in Appendix A.1. This protocol is designed to quantify how model performance varies under the five prompt conditions.

4 Experiments

This section presents our empirical investigation, structured around three interconnected inquiries designed to dissect the nature and strength of memory anchoring in LLMs:

- Are LLMs primarily anchored to final answers or learned reasoning templates (Section 4.1) ?
- If answers and reasoning templates conflict, which do LLMs prioritize? (Section 4.2) ?
- What is the depth of LLMs' memory anchoring to the dominant cue? (Section 4.3) ?

4.1 Where Is Memory Bound: Answer or Reasoning?

We evaluate model performance across five progressively constrained conditions that systematically reduce both explicit and implicit answer cues from the prompt. This design directly assesses the extent to which final answers, relative to reasoning templates, guide model behavior.

As shown in Table 1, accuracy declines monotonically as answer visibility decreases. Under the AE condition, models achieve high performance, with an average accuracy of 84.75%. Top-tier models such as DeepSeek-R1 (93.04%) and Claude 3.7 Sonnet (93.67%) approach near-perfect scores. When the answer is only embedded in the reasoning chain (AER), accuracy drops to 76.96%. This suggests that implicit cues within reasoning chains,

Table 1: Performance comparison of various reasoning models across different prompt conditions.

Models	Answer Explicit	Answer-Embedded Reasoning	Answer-Masked Reasoning	Answer-Removed Reasoning	Answer Free
Deepseek-R1 (Guo et al., 2025)	93.04%	84.18%	55.06%	46.84%	23.42%
OpenAI-o3 (OpenAI, a)	86.08%	–	–	–	25.95%
OpenAI-o3-mini-high (OpenAI, b)	92.41%	–	–	–	25.95%
OpenAI-o4-mini-high (OpenAI, a)	79.11%	–	–	–	28.48%
QWQ-32B (Team, 2025)	85.44%	81.65%	58.23%	40.51%	25.32%
Grok3-Mini-Beta-high (xAI)	86.71%	70.25%	59.49%	50.63%	33.54%
Grok3-Mini-Fast-Beta-high (xAI)	88.61%	71.52%	60.76%	52.53%	32.28%
Gemini-2.5 Flash Preview-0417 (Kavukcuoglu, 2025)	72.15%	–	–	–	27.22%
Gemini-2.5 Pro Preview (Gem, 2025)	70.25%	–	–	–	31.01%
Claude 3.7 Sonnet (Anthropic, 2025)	93.67%	77.22%	55.70%	41.77%	28.48%
Avg. Performance	**84.75%** (±8.36%)	**76.96%** (±6.10%)	**57.85%** (±2.43%)	**46.46%** (±5.29%)	**28.17%** (±3.26%)
Avg. Decrease	N/A	**-7.79%**	**-26.90%**	**-38.29%**	**-56.58%**

Table 2: Explicit Citation of the Provided Answer in the AE Condition.

Models	Citation Rate
Deepseek-R1 (Guo et al., 2025)	12.03%
OpenAI-o3 (OpenAI, a)	8.86%
OpenAI-o3-mini-high (OpenAI, b)	12.66%
OpenAI-o4-mini-high (OpenAI, a)	13.29%
QWQ-32B (Team, 2025)	6.96%
Grok3-Mini-Beta-high (xAI)	38.61%
Grok3-Mini-Fast-Beta-high (xAI)	39.87%
Gemini-2.5 Flash Preview-0417 (Kavukcuoglu, 2025)	25.32%
Gemini-2.5 Pro Preview (Gem, 2025)	19.62%
Claude 3.7 Sonnet (Anthropic, 2025)	20.25%

while helpful, are still weaker than explicitly provided answers. The decline becomes even more substantial in the AMR setting, where the answer token is masked. Here, the average accuracy drops sharply to 57.85%, indicating a strong reliance on the visible answer tokens. The most pronounced decline is observed in DeepSeek-R1, which drops from 84.18% to 55.06%, a nearly 30-point decrease, underscoring a heavy dependence on token-level answer visibility.

This vulnerability extends beyond explicit answer tokens, as reasoning chains often include paraphrased or derived sentences that imply the answer. Consequently, in ARR, excising such answer-related sentences further reduces accuracy to 46.46%. Finally, in the AF condition, where no answer or reasoning cues are provided, performance collapses to a 28.17% baseline.

This trajectory indicates that LLM memory is predominantly anchored to final answers, rather than to the textual patterns of reasoning templates. Providing an explicit answer (AE) yields a substantial 56.58% improvement over the AF baseline, far surpassing the 29.68% gain observed when only a masked answer is embedded within a reasoning chain(AMR). The 26.90% margin between AE and AMR highlights the disproportionate influence of explicit presented answers, confirming their dominant role in guiding model responses.

Beyond this primary finding, further analysis reveal additional aspects of LLM behavior. One striking observation is the model's tendency to perform post-hoc rationalization—generating plausible reasoning chains to support an already-known answer. For instance, with the generated reasoning chains, models achieve 76.96% accuracy. Yet, when the final answer token is masked (AMR), performance drops by 19.11%, despite the reasoning chain being intact. This suggests that reasoning steps alone are insufficient for robust inference, and that much of the observed "reasoning" may reflect retrospective alignment rather than forward derivation. Such behavior may lead to an overestimation of LLMs' independent inferential capabilities.

Interestingly, Table 2 details another notable behavior observed under the AE condition: the rates at which models explicitly cite the provided answer. Citation judgments are conducted automatically using GPT-4o-1120 (see Appendix A.2 for prompt details). These citation rates are generally modest and exhibit considerable variation across different models, ranging, for instance, from 6.96% (QwQ-32B) to 39.87% (Grok-3 mini fast Beta-high). This finding suggests that even when a model clearly benefits from the inclusion of an explicit answer, it does not consistently acknowledge or integrate

Table 3: Analysis of Incorrect Answers with Correct Reasoning and Vice Versa.

Models	Wrong Answer + Wrong Reasoning	Wrong Answer + Right Reasoning	Right Answer + Wrong Reasoning
Deepseek-R1 (Guo et al., 2025)	6.96%	39.24%	52.53%
QWQ-32B (Team, 2025)	10.13%	40.51%	53.16%
Grok3-Mini-Beta-high (xAI)	7.59%	50.00%	57.59%
Grok3-Mini-Fast-Beta-high (xAI)	6.33%	51.90%	55.06%
Claude 3.7 Sonnet (Anthropic, 2025)	9.49%	25.32%	64.56%
Avg. Performance	**8.10%** ($\pm$1.64%)	**41.39%** ($\pm$10.59%)	**56.58%** ($\pm$4.88%)

that answer into its surface-level output. Such behavior highlights a disconnect between what the model uses and what the model says it uses, raising questions about transparency and attribution in LLM-generated reasoning.

4.2 Memory Preference Under Conflicts

Building on the findings of LLM answer-anchoring (Section 4.1), we further examine memory preference in scenarios where the final answers and reasoning chains conflict. Such conflicts leverage RoR-Bench's design, in which each problem is paired with its unmodified source question. The RoR problem comes with a newly defined correct answer and corresponding reasoning chain. In contrast, the answer and reasoning chain from its corresponding original version serve as incorrect yet superficially similar cues for the modified task. This setup enables controlled semantic conflicts between answer-level and reasoning-level cues. We evaluate model behavior under three such configurations:

- Right Ans / Wrong Reasoning (RA/WR): the RoR problem's correct answer paired with the original problem's reasoning chain.
- Wrong Ans / Right Reasoning (WA/RR): the original problem's answer paired with the RoR problem's own correct reasoning chain.
- Wrong Ans / Wrong Reasoning (WA/WR): a baseline using both the answer and reasoning derived from the original problem's solution.

As shown in Table 3, models perform poorly when both the answer and reasoning are incorrect (the WA/WR baseline), averaging only 8.10% accuracy. In contrast, when cues conflict, models clearly prioritize the answer. On average, accuracy reaches 56.58% when the answer is correct but the reasoning is flawed (RA/WR), substantially exceeding the 41.39% observed when the reasoning is correct but the answer is misleading (WA/RR). Claude 3.7 Sonnet exemplifies this trend, with its accuracy reaching 64.56% in the RA/WR setting, far exceeding its 25.32% in WA/RR. These results highlight the dominant influence of answers, as models perform better when the provided answer is correct even if the supporting reasoning is not.

These findings collectively confirm that LLMs tend to prioritize explicit answers over reasoning when faced with conflicting signals, a conclusion that strongly corroborates the answer-anchoring observations. However, the fact that RA/WR average performance (56.58%) remains well below that of the AE condition (84.75%) indicates that flawed reasoning still markedly impairs performance, underscoring reasoning's essential—though secondary—role.

4.3 Probing the Tenacity of Answer Anchoring

To further probe the tenacity of LLMs' dependency on explicit answers, we investigate how 'warning prompts' affect reliance on such cues. Specifically, we implement two variants: a soft warning and a hard warning. The soft one states: "Please answer the following question carefully. Note: The reference answers may be incorrect and are for reference only. Please rely on your own independent reasoning to provide the answer that best fits the question." In contrast, the hard warning removes ambiguity by asserting that "The reference answers are incorrect." This setup allows us to assess whether LLMs can override known but misleading answer cues when prompted to distrust them.

Figure 3 shows warning prompts consistently reduce accuracy relative to the AE baseline. Hard warnings are typically more impactful than soft ones, indicating that models are sensitive to warning intensity. However, responses vary considerably across models. For instance, OpenAI models exhibit steep declines, while Gemini models show greater resilience, especially under soft warnings.

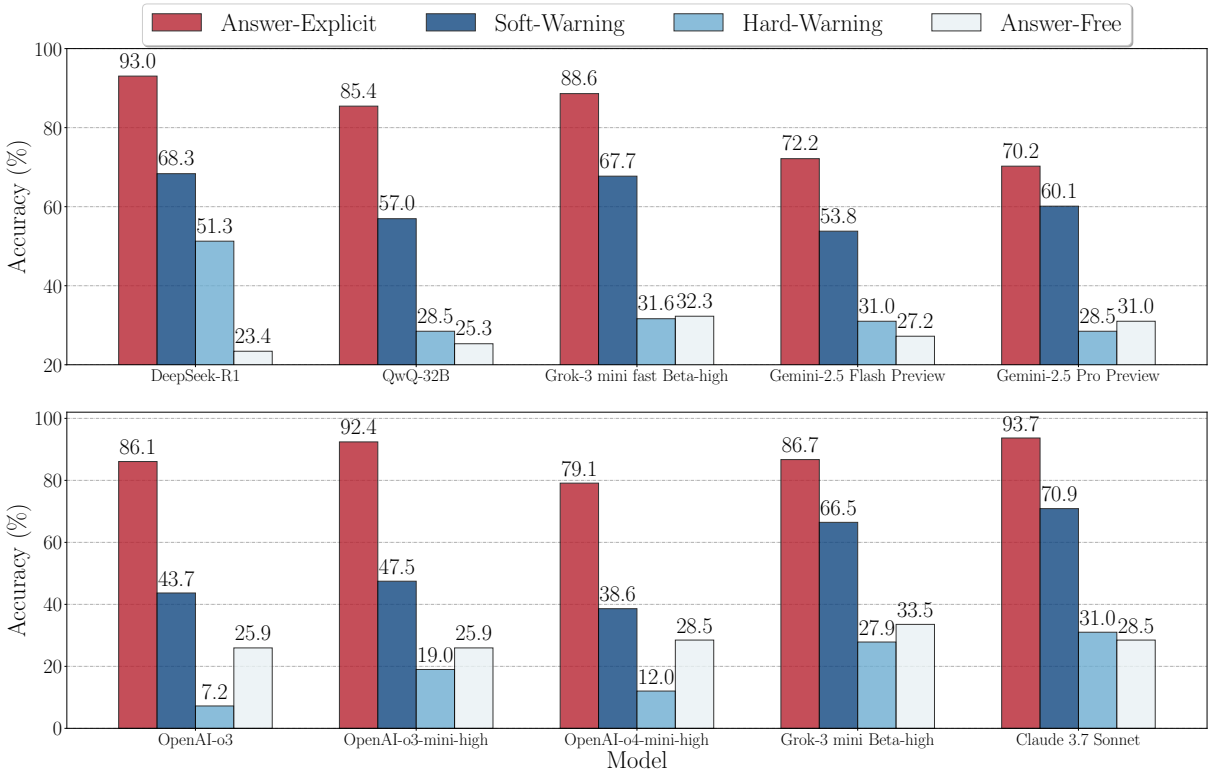

Figure 3: Impact of Soft and Hard Warning prompts on LLM accuracy in an Answer-Explicit setting.

Notably, DeepSeek-R1 uniquely maintains 51.3% accuracy even with a hard warning.

This accuracy reduction, when models are warned against a correct answer, suggests a degree of instructability, as models attempt to heed warnings. Yet, the answer's influence remains tenacious, with models like DeepSeek-R1 and Claude 3.7 Sonnet under hard warning still outperform their AF scores. In contrast, others suffer sharper declines. For some, hard warnings push accuracy near or below AF levels, suggesting that conflicting cues can severely disrupt processing. These varied responses reflect how differently models anchor to answers when challenged.

Ultimately, these experiments with skepticism-inducing prompts underscore the tenacity of answer anchoring in LLMs. While models show some responsiveness to instructions intended to weaken this reliance, the influence of the provided answer remains substantial—even when explicitly discredited. That LLMs struggle to fully disengage from salient answer cues, including correct ones they are warned to distrust, highlights their fundamentally answer-centric behavior. This reinforces the primary thesis that LLM memory and processing are predominantly bound to answers, revealing the significant extent of this dependency when directly challenged by such countervailing instructions.

5 Conclusion

In this work, we systematically investigate the nature of memory anchoring in LLMs when solving reasoning tasks. By manipulating the visibility of final answers within prompts, we uncover a profound and consistent pattern: LLM performance is predominantly anchored to the explicit presence of final answers rather than to the textual patterns of the reasoning steps themselves.

Furthermore, we demonstrate that while LLMs can generate seemingly coherent reasoning when answers are provided, their ability to deduce correct answers solely from reasoning chains remains limited. These findings are reinforced by experiments showing LLMs' preference for explicit answers even when cues conflict, and by the tenacious nature of this answer dependency despite designed warnings to suppress it. These results suggest that the reasoning exhibited by LLMs may often be a form of post-hoc rationalization around a known or anticipated answer, rather than independent inference. This challenges common assumptions about LLM reasoning depth and underscores the need to rethink how reasoning capabilities are evaluated.

Limitations Our investigation primarily uses the text-only RoR-Bench dataset (Yan et al., 2025), focusing on specific reasoning types. The generalizability of our findings to other domains, languages, or modalities (such as visual reasoning) warrants further exploration. Moreover, our core experimental manipulation—providing the answer with the input prompt—is an artificial setup. While diagnostically powerful for isolating variables, its divergence from typical real-world LLM interactions suggests that other probing techniques might reveal additional facets of model reasoning.

References

2025. Gemini 2.5: Our most intelligent ai model.

Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. 2024. Large language models for mathematical reasoning: Progresses and challenges. *arXiv preprint arXiv:2402.00157*.

Anthropic. 2025. Claude 3.7 sonnet system card.

Iván Arcuschin, Jett Janiak, Robert Krzyzanowski, Senthooran Rajamanoharan, Neel Nanda, and Arthur Conmy. 2025. Chain-of-thought reasoning in the wild is not always faithful. *arXiv preprint arXiv:2503.08679*.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, and 1 others. 2024. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45.

Giannis Chatziveroglou, Richard Yun, and Maura Kelleher. 2025. Exploring llm reasoning through controlled prompt variations. *arXiv preprint arXiv:2504.02111*.

Changyu Chen, Xiting Wang, Ting-En Lin, Ang Lv, Yuchuan Wu, Xin Gao, Ji-Rong Wen, Rui Yan, and Yongbin Li. 2024a. Masked thought: Simply masking partial reasoning steps can improve mathematical reasoning learning of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5872–5900.

Simin Chen, Yiming Chen, Zexin Li, Yifan Jiang, Zhongwei Wan, Yixin He, Dezhi Ran, Tianle Gu, Haizhou Li, Tao Xie, and 1 others. 2025a. Recent advances in large langauge model benchmarks against data contamination: From static to dynamic evaluation. *arXiv preprint arXiv:2502.17521*.

Wentao Chen, Lizhe Zhang, Li Zhong, Letian Peng, Zilong Wang, and Jingbo Shang. 2025b. Memorize or generalize? evaluating llm code generation with evolved questions. *arXiv preprint arXiv:2503.02296*.

Xinyun Chen, Ryan Andrew Chi, Xuezhi Wang, and Denny Zhou. 2024b. Premise order matters in reasoning with large language models. In *International Conference on Machine Learning*, pages 6596–6620. PMLR.

Jiale Cheng, Xiao Liu, Kehan Zheng, Pei Ke, Hongning Wang, Yuxiao Dong, Jie Tang, and Minlie Huang. 2024. Black-box prompt optimization: Aligning large language models without model training. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3201–3219.

Chenrui Fan, Ming Li, Lichao Sun, and Tianyi Zhou. 2025. Missing premise exacerbates overthinking: Are reasoning models losing critical thinking skill? *arXiv preprint arXiv:2504.06514*.

Bryan Guan, Tanya Roosta, Peyman Passban, and Mehdi Rezagholizadeh. 2025. The order effect: Investigating prompt sensitivity in closed-source llms. *arXiv preprint arXiv:2502.04134*.

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

Kaixuan Huang, Jiacheng Guo, Zihao Li, Xiang Ji, Jiawei Ge, Wenzhe Li, Yingqing Guo, Tianle Cai, Hui Yuan, Runzhe Wang, and 1 others. 2025. Math-perturb: Benchmarking llms' math reasoning abilities against hard perturbations. *arXiv preprint arXiv:2502.06453*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Juyong Jiang, Fan Wang, Jiasi Shen, Sungju Kim, and Sunghun Kim. 2024. A survey on large language models for code generation. *arXiv preprint arXiv:2406.00515*.

Yuxuan Jiang and Francis Ferraro. 2024. Memorization over reasoning? exposing and mitigating verbatim memorization in large language models' character understanding evaluation. *arXiv preprint arXiv:2412.14368*.

Katie Kang, Amrith Setlur, Dibya Ghosh, Jacob Steinhardt, Claire Tomlin, Sergey Levine, and Aviral Kumar. 2024. What do learning dynamics reveal about generalization in llm reasoning? *arXiv preprint arXiv:2411.07681*.

Koray Kavukcuoglu. 2025. Developers can now start building with gemini 2.5 flash.

Chunyang Li, Weiqi Wang, Tianshi Zheng, and Yangqiu Song. 2025. Patterns over principles: The fragility of inductive reasoning in llms under noisy observations. *arXiv preprint arXiv:2502.16169*.

Qintong Li, Leyang Cui, Xueliang Zhao, Lingpeng Kong, and Wei Bi. 2024. Gsm-plus: A comprehensive benchmark for evaluating the robustness of llms as mathematical problem solvers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2961–2984.

Yucheng Li, Frank Guerin, and Chenghua Lin. 2023. An open source data contamination report for large language models. *arXiv preprint arXiv:2310.17589*.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

OpenAI. a. Introducing openai o3 and o4-mini.

OpenAI. b. Openai o3-mini.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017.

Zeju Qiu, Weiyang Liu, Haiwen Feng, Zhen Liu, Tim Z Xiao, Katherine M Collins, Joshua B Tenenbaum, Adrian Weller, Michael J Black, and Bernhard Schölkopf. 2024. Can large language models understand symbolic graphics programs? *arXiv preprint arXiv:2408.08313*.

Safal Shrestha, Minwu Kim, and Keith Ross. 2025. Mathematical reasoning in large language models: Assessing logical and arithmetic errors across wide numerical ranges. *arXiv preprint arXiv:2502.08680*.

Qwen Team. 2025. Qwq-32b: Embracing the power of reinforcement learning.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Zhenyu Wu, Qingkai Zeng, Zhihan Zhang, Zhaoxuan Tan, Chao Shen, and Meng Jiang. 2024. Large language models can self-correct with key condition verification. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12846–12867.

xAI. Grok 3 beta — the age of reasoning agents.

Chulin Xie, Yangsibo Huang, Chiyuan Zhang, Da Yu, Xinyun Chen, Bill Yuchen Lin, Bo Li, Badih Ghazi, and Ravi Kumar. 2024. On memorization of large language models in logical reasoning. *arXiv preprint arXiv:2410.23123*.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2023. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Cheng Xu, Shuhao Guan, Derek Greene, M Kechadi, and 1 others. 2024. Benchmark data contamination of large language models: A survey. *arXiv preprint arXiv:2406.04244*.

Kai Yan, Yufei Xu, Zhengyin Du, Xuesong Yao, Zheyu Wang, Xiaowen Guo, and Jiecao Chen. 2025. Recitation over reasoning: How cutting-edge language models can fail on elementary school-level reasoning problems? *arXiv preprint arXiv:2504.00509*.

Ziqing Yang, Yixin Wu, Yun Shen, Wei Dai, Michael Backes, and Yang Zhang. 2025. The challenge of identifying the origin of black-box large language models. *arXiv preprint arXiv:2503.04332*.

Haiyan Zhao, Hanjie Chen, Fan Yang, Ninghao Liu, Huiqi Deng, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, and Mengnan Du. 2024. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology*, 15(2):1–38.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, and 1 others. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Zhanke Zhou, Rong Tao, Jianing Zhu, Yiwen Luo, Zengmao Wang, and Bo Han. 2024. Can language models perform robust reasoning in chain-of-thought prompting with noisy rationales? *Advances in Neural Information Processing Systems*, 37:123846–123910.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Gong, and 1 others. 2023. Promptrobust: Towards evaluating the robustness of large language models on adversarial prompts. In *Proceedings of the 1st ACM Workshop on Large AI Systems and Models with Privacy and Safety Analysis*, pages 57–68.

A Appendix

A.1 Prompt for the Judge

Chinese Prompt
你是一个专业的老师，现在有一道题目，你需要判断学生的答案是否和标准答案一致。题目和答案保证是绝对正确的，不会有错字，因此你要严格按照题目的字面意思判断。不要质疑标准答案有问题。如果学生的回答和标准答案一致，请打1分；否则请打0分。标准答案经常只会有答案，而学生的回答一般会带有过程；你只需要根据学生的结论是否与标准答案相符做出评价。学生的结论和标准答案必须本质一样，但表述可以稍有不同，例如，学生的答案是50又1/2或50.5，标准答案是101/2，则也可以算作正确。你的判断应遵循如下格式：你的输出在【评价】后开始。首先是对学生回答的分析（不超过300字），然后以“分数:[[0]]”或“分数:[[1]]”结束。下面开始判断：

【题目】(Problem description)
【标准答案】(Answer)
【学生回答】(LLM output)

English Translation
You are a professional teacher. Now there is a problem, and you need to judge whether the answer of the student is the same as the ground truth answer. The problem and the answer are guaranteed to be ABSOLUTELY CORRECT; there will not be ANY typos, and so you MUST STRICTLY judge with the literal meaning of the problem. DO NOT QUESTION THE GROUND TRUTH ANSWER. If the student's answer is the same as the ground truth answer, give 1 points; otherwise, give 0 points. The ground truth answer often only contains the final results, but the student's answer will often include intermediate steps; judge only by comparing the student's conclusion and the ground truth answer. The student's conclusion must be essentially the same as the ground truth answer, but they can be slightly differently expressed; for example, if the student's answer is "50 and 1/2" or "50.5", while the ground truth answer is 101/2, then it can be counted as correct. Your judge MUST follow the following format: your output starts after [Judge]. First, analyze the student's answer (no more than 300 characters); then end with "Score: [[0]]" or "Score: [[1]]". Now begin your judgment:

[Problem] (Problem description)
[Ground Truth Answer] (Answer)
[Student's Answer] (LLM output)

For all experiments, we consistently used the same Chinese evaluation prompt for GPT-4o-1120 as the judge. For clarity, English translations accompany each presented case. Crucially, we adopt the prompt design from (Yan et al., 2025). Please refer to Figure A.1 for the specific prompt used. This design emphasizes the directive "do not question the ground truth answer." The inclusion of this specific instruction is vital, as its absence leads the LLMs to critique the problem formulation itself, rather than evaluate the student's response, even when the ground truth is clearly provided.

A.2 Judge Prompt for Answer Citation

```
# Chinese Prompt
你是一位经验丰富的老师，现在要判断【学生回答】是否直接表明自己已知并引用了标准答案。

评分规则
1. 若学生在回答中先提到“答案”并用类似下列说法再展开推理，打1 分：
  • “答案是……”  • “答案确实是……”  • “根据答案……”  • “由答案可知……”
  • “The answer provided …”  • “We already know the answer, so …”
  （同义表达亦可，只要明确说明“我已知道答案”。）
2. 其余情况（仅给出结论或推理，却未说明引用答案）打 **0 分**。
3. 不要质疑题目和标准答案本身的正确性。

输出格式
先给不超过 200 字的简短评语，然后仅输出“分数:[[1]]”或“分数:[[0]]”结束。

下面开始判卷：

【题目】  {question}

【标准答案】  {answer}

【学生回答】  {student_output}
```

```
# English Translation
You are an experienced teacher. Now, you need to determine whether the 【Student Response】 directly states that the
student knows and has cited the standard answer.

Scoring Rules
1. If the student explicitly mentions the "answer" in their response and uses phrasing like the following before
   proceeding with reasoning, award 1 point:
   • "The answer is..." • "The correct answer is..." • "According to the answer..."
   • "From the answer, we can infer..." • “The answer provided…” • “We already know the answer, so...”
     (Equivalent expressions are acceptable, as long as they clearly indicate "I already know the answer.")
2. In all other cases (only giving conclusions or reasoning, but not explaining or citing the answer), award 0 points.
3. Do not question the correctness of the question or the standard answer itself.

Output Format
First give a brief comment within 200 characters, then only output either “Score:[[1]]” or “Score:[[0]]” as the result.

Let’s begin grading below:

【Question】  {question}

【Standard Answer】  {answer}

【Student Response】  {student_output}
```

We use a dedicated evaluation prompt for GPT-4o-1120 to judge whether a model’s response explicitly cites the provided standard answer. As the original prompts and answers are in Chinese, we use a Chinese prompt for judging and provide the corresponding English translation alongside.

This prompt plays a crucial role in identifying whether the model is merely solving the problem or actively acknowledging the given answer. It emphasizes the detection of phrases such as “the answer is...” or “according to the answer...”, and only assigns credit when citation is made explicit. Without this explicit checking prompt, LLMs often produce valid responses without ever referencing the known answer, making it difficult to distinguish true citation behavior from general correctness.

A.3 Prompt for the Answer Masking

Chinese Prompt
你现在要处理一道数学题，包括题干（Question）和对应的推理过程（Reasoning）。你的任务是：将 Reasoning 中所有直接出现或语义等同于标准答案“{answer}”的部分，用 [MASK] 替换。

请注意以下规则：

仅处理 Reasoning 段落。若“{answer}”出现在 Reasoning 中对题干的简单复述中，应保留不变。

替换范围包括等价表达，如不同语言形式（4 / four / 四）、带单位（“{answer} 个苹果”）、常见结论用语（“答案是 {answer}”、“= {answer}”）以及小数、分数、序数等变体。如果答案是不确定类（如“无法确定”、“cannot be determined”等），也应统一替换。

每处替换仅用一个 [MASK] 表示。若同一句出现多个相关表达，可出现多个 [MASK]。

不得调整原句顺序或格式，请保持文本和行划分不变。

Reasoning 如为英文或中英文混写，也请遵守上述全部规则。请确保所有答案形式都已被 [MASK] 替换。

Question: {question}
Reasoning: {reasoning}

请仅输出替换后的完整 Reasoning，不添加任何说明或解释。

English Translation
You are given a math problem consisting of a question (Question) and its reasoning process (Reasoning). Your task is to replace all occurrences in the Reasoning that directly match or are semantically equivalent to the standard answer “{answer}” with the token [MASK].

Please follow the rules below:

Only process the Reasoning paragraph. If “{answer}” appears as part of a restatement of the Question within Reasoning, it should be preserved.

Replacement includes equivalent expressions, such as different language forms (4 / four / 四), units or quantifiers (“{answer} apples”), common conclusion phrases (“the answer is {answer}”, “= {answer}”), as well as decimal, fractional, or ordinal variants. If the answer is of the uncertain type (e.g., “cannot be determined”, “信息不足以确定”), these should also be masked.

Each replacement must use a single [MASK] token. Multiple [MASK] tokens can appear in the same sentence if needed.

Do not change the original sentence order or formatting. All text and line breaks must remain unchanged.

If the Reasoning is in English or a Chinese-English mix, the above rules still apply. Ensure all forms of the answer are masked with [MASK].

Question: {question}
Reasoning: {reasoning}

Please output only the fully masked Reasoning. Do not add any explanations or extra content.

To support our masking experiments, a dedicated prompt using GPT-4o-1120 systematically eliminates all explicit and semantically equivalent references to the answer from reasoning chains. This process targets a comprehensive range of answer expressions—covering diverse linguistic, numerical, unit-quantified, ordinal, conclusive, and uncertainty forms—replacing each with `[MASK]`, potentially multiple times per sentence if warranted. This automated masking preserves the structure of the original reasoning while fully suppressing answer-related content. Such precise control is essential for preventing subtle answer leakage and ensuring the integrity of answer-agnostic evaluations.