

MME-REASONING: A COMPREHENSIVE BENCHMARK FOR LOGICAL REASONING IN MLLMs

Anonymous authors

Paper under double-blind review

ABSTRACT

Logical reasoning is a fundamental aspect of human intelligence and an essential capability for multimodal large language models (MLLMs). Despite the significant advancement in multimodal reasoning, existing benchmarks fail to comprehensively evaluate their reasoning abilities due to the lack of explicit categorization for logical reasoning types and an unclear understanding of reasoning. To address these issues, we introduce **MME-Reasoning**, a comprehensive benchmark designed to evaluate the reasoning ability of MLLMs, which covers all three types of reasoning (*i.e.*, inductive, deductive, and abductive). We carefully curate the data to ensure that each question effectively evaluates reasoning ability rather than perceptual skills or knowledge breadth, and extend the evaluation protocols to cover the evaluation of diverse questions. Our evaluation reveals substantial limitations of SoTA MLLMs when subjected to holistic assessments of logical reasoning capabilities. Even the most advanced MLLMs show limited performance in comprehensive logical reasoning, with notable performance imbalances across reasoning types. In addition, we conducted an in-depth analysis of approaches such as “thinking mode” and Rule-based RL, which are commonly believed to enhance reasoning abilities. We hope the community can pay more attention to the comprehensive reasoning capabilities of MLLMs instead of only focusing on its subset, such as Math.

1 INTRODUCTION

Logical reasoning (Liu et al., 2025a), a fundamental cognitive process of analyzing premises and evidence to reach valid conclusions, serves as the cornerstone of human intelligence. Multimodal reasoning (Jaech et al., 2024) enables humans to integrate information from different modalities, such as visual and text, which is essential for tackling complex tasks. Recently, with the emergence of reasoning large language models (LLMs) (Dubey et al., 2024) such as DeepSeek-R1 (DeepSeek-AI, 2025), injecting reasoning capability into MLLMs (OpenAI, 2024; Qwen Team, 2025a) has begun to be explored (Peng et al., 2025b). Despite the significant progress in reasoning MLLMs, a comprehensive evaluation of their capabilities still remains an open challenge. Therefore, it is important to establish a [comprehensive](#) evaluation benchmark to assess the reasoning capabilities of MLLMs and further accelerate the development of this field.

Currently, most benchmarks (Fu et al., 2023; Wang et al., 2024a; Lu et al., 2023; Yue et al., 2024a;b; Gong et al., 2024; He et al., 2024) designed for multimodal reasoning primarily focus on knowledge-driven tasks. For example, MathVista (Lu et al., 2023) and MathVerse (Zhang et al., 2024) provide comprehensive evaluations of MLLMs’ mathematical reasoning abilities. OlympiadBench (He et al., 2024) and EMMA (Hao et al., 2025) expand the scope to include additional subjects, such as physics and chemistry. Apart from knowledge-driven tasks, some works (Chia et al., 2024; Zhang et al., 2025b) have begun to decouple knowledge from reasoning, aiming to assess the reasoning abilities of MLLMs independent of domain knowledge. For instance, VisualPuzzles (Song et al., 2025) focus on reasoning-focused, knowledge-light tasks.

Despite recent advances, existing benchmarks still suffer from several problems as outlined below, [which limits a more comprehensive evaluation of multimodal reasoning](#). (1) **Lacking explicit categorization of reasoning and insufficient coverage of reasoning types**. Logical reasoning is typically classified into three types: **abduction, deduction, and induction** (Peirce, 2014). Most existing benchmarks primarily concentrate on evaluating MLLMs’ inductive and deductive reasoning

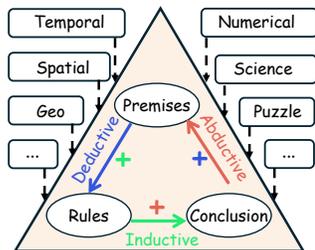


Figure 1: Reasoning Types.

Table 1: Comparison with other benchmarks.

Benchmark	Size	Question Types	Reasoning Types
MathVista	6.1K	MCQ, Free-form	Ind., Ded.
LogicVista	0.4K	MCQ	Ind., Ded.
VisualPuzzles	1.2K	MCQ	Ind.
VisuLogic	1K	MCQ	Ind.
MME-Reasoning	1.2K	MCQ, Free-form Rule-based	Ind., Ded., Abd.

ability. For example, most of the questions in MathVerse (Lu et al., 2023) belong to deductive reasoning, which uses rules and premises to derive conclusions. PuzzleVQA (Chia et al., 2024) only contains questions of inductive reasoning, which learns rules based on premises and conclusions. However, abductive reasoning ability (*i.e.*, exploring premises to explain a conclusion based on the conclusion and rules) is rarely evaluated. (2) **The concept of reasoning is not clear enough**, which is reflected in confusing perception with reasoning or equating reasoning with the complexity of the required knowledge. For example, MathVista (Lu et al., 2023) contains many questions that can be answered through visual perception, while OlympiadBench (He et al., 2024) includes questions that require advanced domain knowledge, which the model may not have access to. This may lead to an inaccurate evaluation of MLLMs’ reasoning ability.

To address these and [construct a comprehensive benchmark for multimodal reasoning](#), we introduce MME-Reasoning, a comprehensive benchmark specifically designed to evaluate the reasoning capability of MLLMs. MME-Reasoning consists of 1,188 carefully curated questions that systematically cover types of logical reasoning (*i.e.*, inductive, deductive, and abductive), while spanning a range of difficulty levels, as illustrated in Fig. 2. Besides, we identify 5 key abilities related to multimodal reasoning, including calculation, planning and exploring, spatial-temporal, pattern analysis, and causal chaining analysis, and annotate the type of ability assessed by each question. To ensure a true evaluation of reasoning ability, MME-Reasoning eliminates questions that can be answered purely through perception or require complex domain knowledge, thereby focusing on the core reasoning skills of the model. Furthermore, MME-Reasoning incorporates a variety of evaluation methods, including multiple-choice, free-form, and rule-based (*e.g.*, Sudoku Puzzles) questions. Employing multiple evaluation methods enables a wider variety of question types, thereby facilitating a more comprehensive evaluation of models’ capabilities. Experiments on diverse SoTA MLLMs (*i.e.*, chat/thinking, open/closed-source) reveal several key findings:

- **MLLMs exhibit significant limitations and pronounced imbalances in reasoning capabilities.** Even the most advanced MLLMs achieve only limited results under holistic logical reasoning evaluation, with Gemini-Pro-2.5-Thinking scoring only 60.19%, followed by Seed1.5-VL (59.85). These results indicate that MME-Reasoning, through its comprehensive evaluation of all the logical reasoning types, establishes a systematic and challenging benchmark for multimodal reasoning.
- **Abductive reasoning remains a major bottleneck for current MLLMs.** While most models demonstrate competent deductive reasoning, their abductive reasoning lags significantly. Closed-source models exhibit an average gap of 5.38 points between deductive and abductive tasks, which further widens to 9.81 among open-source models. Since it underpins many real-world tasks, addressing this gap is crucial for improving overall reasoning.
- **Reasoning length scales with task difficulty, benefiting performance but accompanied by marginal effects and decreasing token efficiency.** Thinking Models exhibit longer reasoning chains, particularly on more difficult questions, demonstrating adaptive inference budgeting and enhanced depth of reasoning. A positive correlation between average token count (ATC) and accuracy supports the effectiveness of extended outputs, especially in complex tasks. However, this performance gain plateaus beyond a certain length, revealing diminishing returns.

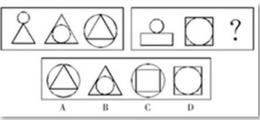
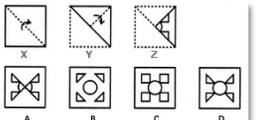
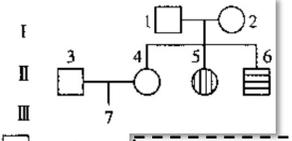
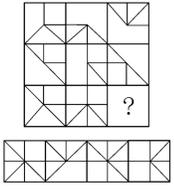
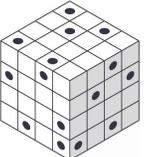
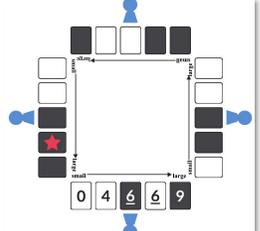
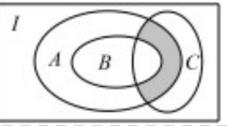
Inductive	Deductive	Abductive
 <p>Q: Observe the following graphic pattern and choose the option that best fits the logic from A, B, C, D to fill in the question mark (?).</p>	 <p>Q: Fold and subtract parts along the dotted lines shown in the figure, and choose the shape that is most similar to the unfolded shape of Figure Z among A, B, C, and D.</p>	 <p>Q: The gene for disease A is represented by A or a, and the gene for disease B is represented by E or e. One of these diseases has a gene locus on the X chromosome. What's the probability that individual 4 has the same genotype as individual 2?</p>
 <p>Q: Choose the most appropriate option from the given four choices to fill in the question mark, so that it presents a certain regularity</p>	 <p>Q: Stack 64 small cubes to form a large cube. Punch holes in the part marked with black dots, as shown in the picture below. How many small cubes are pierced?</p>	 <p>Q: The rules of the game are as follows: (1) The player who gets the cards arranges them in front of themselves in ascending order from left to right. ... what are the numbers on the cards marked by star?</p>
 <p>Q: Find the arrangement pattern of numbers in the pyramid and calculate the value of A, B, and C.</p>	 <p>Q: I is the universal set, and A, B, and C are its subsets. Which set does the shaded region represent?</p>	

Figure 2: Example of questions in MME-Reasoning which covers comprehensive reasoning types.

2 RELATED WORKS

2.1 MULTIMODAL REASONING BENCHMARKS

Early works such as CLEVR (Johnson et al., 2016) assess compositional and spatial reasoning, while more recent benchmarks (Lu et al., 2024; Zhang et al., 2025b; Chia et al., 2024) focus on symbolic logic reasoning. EMMA Hao et al. (2025) focuses on ensuring that questions cannot be answered by LLMs using only the image caption. VisualPuzzles Hu et al. (2024), MM-IQ Cai et al. (2025), and VisuLogic Xu et al. (2025) try to evaluate MLLMs' reasoning capabilities through puzzle-solving. Rbench-V Guo et al. (2025b) is designed to evaluate models' capability of MCot (i.e., whether a model can perform reasoning visually). However, these benchmarks typically focus on narrow subtypes and fail to offer a comprehensive evaluation across deductive, inductive, and abductive paradigms. To evaluate the reasoning capability, some benchmarks (Rein et al., 2023; He et al., 2024; Jiang et al., 2025) improve the complexity of questions. However, most of them overly depend on domain-specific knowledge rather than logical inference. Evaluation protocols are also limited—most rely on multiple-choice formats and lack support for open-ended or rule-based assessment. In contrast, our benchmark provides a fine-grained evaluation of visual reasoning, explicitly covering the three classical reasoning types.

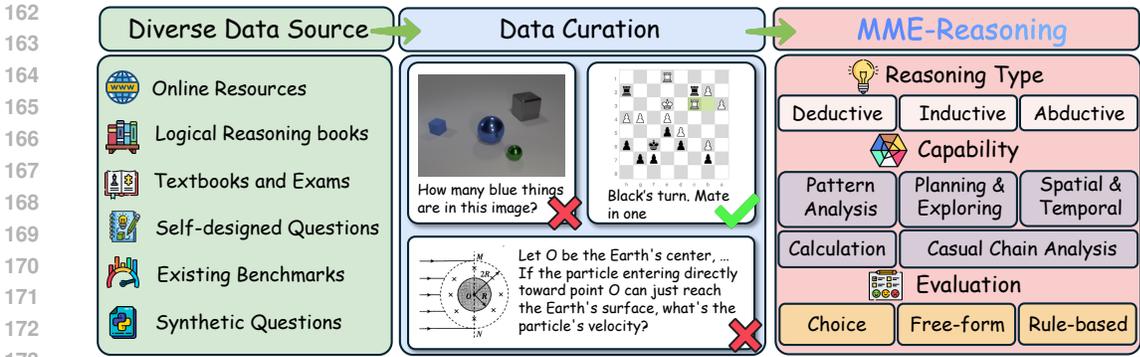


Figure 3: The overall construction process of MME-Reasoning.

2.2 MULTIMODAL REASONING

Chain-of-thought (CoT) reasoning (Wei et al., 2022) has emerged as a key paradigm for enhancing the reasoning capability of LLMs, enabling more transparent and accurate decision-making in complex tasks. Inspired by its success in text-only settings, CoT has recently been extended to MLLMs, giving rise to multimodal chain-of-thought (MCoT) reasoning (Jiang et al., 2024; Zhang et al., 2023; Chen et al., 2023; Peng et al., 2024; Lu et al., 2025; Xia et al., 2024). Early approaches such as Multimodal-CoT (Zhang et al., 2023) and IPVR (Chen et al., 2023) demonstrate that generating intermediate reasoning steps significantly improves model performance in visual question answering. Other methods (Yao et al., 2023; Zheng et al., 2024; Hu et al., 2024) introduce graph structures, debating agents, and visual intermediate states to further enhance interpretability and reasoning depth. More recently, following the success of Deepseek-R1, the Group Relative Policy Optimization (GRPO) algorithm has gained traction in the development of multimodal models. Several works (Meng et al., 2025; Zhou et al., 2025; Yingzhe et al., 2025; Chen et al., 2025; Feng et al., 2025) adapt GRPO to solve mathematical tasks, demonstrating promising reasoning capabilities. Other works (Shen et al., 2025; Liu et al., 2025b) apply GRPO to enhance visual competencies such as grounding.

3 THE MME-REASONING BENCHMARK

We introduce MME-Reasoning, a comprehensive benchmark for evaluating the reasoning ability of MLLMs. MME-Reasoning consists of 1,188 questions (1,008 newly collected items). MME-Reasoning covers three types of reasoning (*i.e.*, inductive, deductive, abductive) and includes three question types (*i.e.*, MCQ, free-form, rule-based). We further divided MME-Reasoning into three difficulty levels (*i.e.*, easy, medium, hard). The key statistics are shown in Tab. 2 and the construction pipeline of MME-Reasoning is shown in Fig. 3.

3.1 DESIGN PRINCIPLES OF MME-REASONING

To ensure a comprehensive evaluation of multimodal reasoning, MME-Reasoning is guided by the following principles: **1) *Comprehensiveness***. According to *Charles Sanders Peirce’s classification of reasoning*, deduction, induction, and abduction can be distinguished based on different arrangements of rule, case, and result. Therefore, a comprehensive evaluation of reasoning ability should include all three types of reasoning tasks. **2) *Beyond Perception***. Each question should be carefully designed to ensure that the answer is obtained through a reasoning process instead of simple visual recognition. **3) *Minimizing Knowledge Reliance***. It is essential to ensure that the questions do not require complex domain knowledge, thereby preventing models from being penalized for the absence of specialized information. In MME-Reasoning, the domain expertise is limited to K12 or below. **4) *Diverse evaluation formats***. The benchmark should consist of diverse question types, avoiding incomplete evaluation caused by a narrow range of task types.

Table 2: Statistics of MME-Reasoning.

Statistics	Number
Total	1188 (100%)
- Newly-add questions	84.85%
- Sampled questions	15.15%
Question Type	
- Multi-choice questions	58.50%
- Free-form questions	31.57%
- Rule-based questions	9.93%
Image Type	
- Single-image questions	58.50%
- Multi-image questions	31.57%
Disciplinary	
- Disciplinary questions	31.48%
- Non-discipl. questions	68.52%

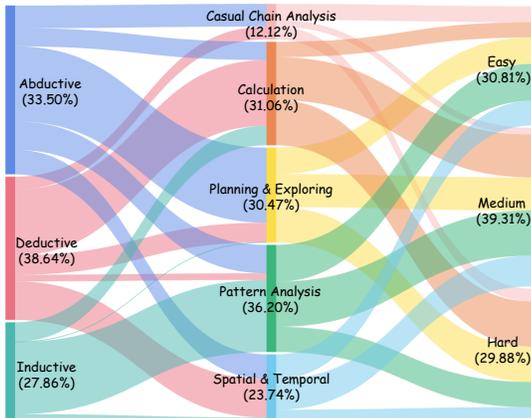


Figure 4: Overview of MME-Reasoning.

3.2 DATA COLLECTION AND CURATION

Data Collection. We collect questions related to multimodal reasoning from a variety of sources, including 1) *Textbooks* can provide subject exam questions (e.g., mathematics, physics, chemistry, and biology). 2) *Online resources, books on logical practice, and Chinese Civil Service Examination (Logic Test)* primarily includes IQ test questions, logic games (e.g., Mate-in-one), and other tasks highly related to logical reasoning. 3) *Synthetically generated questions.* Some visual reasoning problems, such as Number Bridge and Sudoku, can be generated based on specific rules. We develop code to generate a wide variety of such logic puzzles, covering different types and a range of difficulty levels. 4) *Questions from existing benchmarks.* We sample 80 questions from PuzzleVQA (Chia et al., 2024) and 100 questions from MMIQ (Cai et al., 2025), excluding questions based on shape size identification which can be directly answered based on visual perception. 5) *Self-designed questions.* We mainly construct questions related to spatial and temporal reasoning. The spatial reasoning questions involve relative spatial relationships and navigation tasks, with the question design methodology inspired by VSIBench (Yang et al., 2024). For temporal reasoning, the questions mainly focus on sequence judgment. We sample frames from videos in YouCook2 (Zhou et al., 2018) and VideoMME (Fu et al., 2024) as the sources of images. Note that for questions with well-defined rules such as Maze, we include the corresponding rules as part of each question. The composition of MME-Reasoning is shown in Fig. 4 and details about the question source and type are listed in the Appendix.

Data Curation. We initially collect around 4k questions from various sources mentioned above. Following the design principles of MME-Reasoning, we conduct a careful manual curation process to ensure the quality of the benchmark. Specifically, we exclude questions that depend solely on visual recognition, require complex domain-specific knowledge, too easy to evaluate the reasoning ability. This curation process ensures that the remaining questions are well-aligned with our goal of evaluating visual reasoning ability, rather than perceptual skills or the breadth of specialized knowledge. For questions with multiple possible answers, we first try to convert them into rule-based (will be introduced in Sec. 4.2) or multiple-choice questions; otherwise, discard them. Finally, to comprehensively evaluate the multimodal reasoning ability, we balance the distribution of questions across the three reasoning types. This approach prevents the benchmark from being overly biased towards evaluating the ability of any single reasoning type. Through this data curation process, we filter 1,008 questions from the initially collected questions.

Metadata Annotation. Further, we annotate questions in MME-Reasoning with information including question type (i.e., multiple-choice, free-form, and rule-based), difficulty (i.e., easy, medium, hard), capability (i.e., pattern analysis, planning and exploring, spatial and temporal, calculation, casual chain analysis), and reasoning type (i.e., deductive, inductive, and abductive). The specific rules are listed in our Appendix.

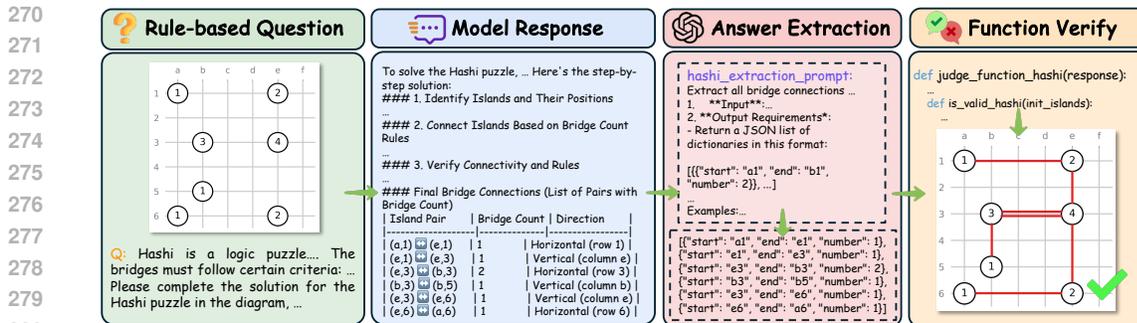


Figure 5: Evaluation of rule-based questions.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

We conduct extensive evaluations on SoTA MLLMs including: **(1) Thinking Models.** We first evaluate thinking MLLMs that focus on improving the models’ multimodal reasoning which can be divided into *Close-source models* including (1) o1 (Jaech et al., 2024), and o4-mini (, 2025); (2) Gemini-2.5-Flash-Thinking and Gemini-2.5-Pro-Thinking (Gemini et al., 2023); (3) Claude-3.7-Sonnet-Thinking, Claude-4-Sonnet-Thinking (Anthropic, 2022); (4) Seed1.5-VL-Thinking (Guo et al., 2025a); and *Open-source models* including (1) QvQ-72B-Preview (Team, 2024); (2) Kimi-VL-A3B-Thinking (Team et al., 2025b); (3) LlamaV-o1 (Thawakar et al., 2025); (4) Virgo-72B (Du et al., 2025). **(2) Chat Models.** *Close-source models:* (1) GPT-4o (OpenAI, 2024); (2) Claude-3.7-Sonnet (Anthropic, 2022) (3) Kimi-latest (Team et al., 2025a); (4) Seed1.5-VL (Guo et al., 2025a). *Open-source models:* (1) Qwen-2.5-VL (7B, 32B, 72B) (Qwen Team, 2025a); (2) InternVL-3 (8B, 38B, 78B) (Zhu et al., 2025); (3) LLaVA-Onevision-72B (Li et al., 2024); (4) Molmo (7B-O, 7B-D, 72B) (Deitke et al., 2024); (5) Kimi-VL-A3B-Instruct (Team et al., 2025b). **(3) Rule-based RL Models.** Rule-based RL is a highly promising strategy for eliciting reasoning paradigms in models. Therefore, we further evaluated MLLMs trained using Rule-based RL, including: (1) R1-VL (Zhang et al., 2025a), (2) R1-Onevision (Yang et al., 2025), (3) Vision-R1 (Huang et al., 2025), (4) MM-Eureka (7B, 32B) (Meng et al., 2025), (5) VL-Rethinker (7B, 72B) (Wang et al., 2025). We use GPT-4o-mini to extract answers. Due to rate limits, we sample 302 questions to construct mini-set with the same distribution for o1’s evaluation, all other models are evaluated on the entire benchmark.

4.2 EVALUATION PROTOCOLS

Following MathVista (Lu et al., 2023), the evaluation consists of two steps: extracting answers and judging answers. For different types of questions, we designed specific prompts for GPT to extract answers that are composed of extraction rules and examples. For MCQ questions, we match the extracted answers with the reference answers. For free-form questions, we use GPT to judge the consistency between the extracted answers and the reference answers. For rule-based questions, we first use GPT to extract answers and convert them into an intermediate format, which is then judged using specific scripts. For example, in a Number Bridge problem, we first use GPT to extract the start and end points of each bridge, then convert the answers into a specific matrix format, and finally determine correctness based on predefined rules. An example of rule-based evaluation is shown in Fig. 5.

4.3 MAIN RESULTS

MME-Reasoning poses significant challenges for vision-language reasoning. As shown in Tab. 3, the best-performing model, Gemini-2.5-Pro-Thinking, achieved an average score of 60.2%. The latest MLLM, Seed1.5-VL, achieved a comprehensive score of 59.9. Qwen2.5-VL and Claude-3.7-Sonnet achieved scores of 35.9 and 57.2 on OlympiadBench, yet only reached 34.1 on MME-Reasoning. These results indicate that the benchmark sets stringent standards for evaluating models’ logical reasoning capabilities by comprehensively assessing three distinct reasoning types.

Table 3: Performance comparison of SoTA MLLMs. The top three are highlighted in blue. † indicates the model was evaluated on the mini-set. “T” represents “Thinking”. CAL., P&E., PA., S&T., and CCA. represent calculation, plan and exploration, pattern analysis, spatial and temporal, and casual chain analysis.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P&E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
<i>Human Performance</i>									
Human Expert [†]	75.0	84.4	84.9	80.3	88.1	85.8	76.9	85.6	83.4
<i>Close-source & Thinking</i>									
Gemini-2.5-Pro-T	68.0	64.4	53.7	52.1	90.3	64.0	51.7	62.8	60.2
Seed1.5-VL-T	67.2	62.7	56.0	47.2	82.6	64.5	52.3	60.8	59.9
o4-mini	63.1	58.3	57.2	50.4	59.0	60.6	51.4	59.0	57.5
o1 [†]	50.0	38.5	41.5	43.7	52.4	50.8	42.3	42.3	45.7
Claude-4-Sonnet-T	33.3	35.9	33.0	36.2	47.9	39.4	32.0	35.7	36.1
Claude-3.7-Sonnet-T	30.4	27.6	32.3	38.3	46.5	34.6	36.2	31.7	34.1
Gemini-2.5-Flash-T	19.8	21.3	20.9	33.0	38.9	28.1	22.1	24.6	25.2
<i>Close-source & Chat</i>									
Seed1.5-VL	52.0	42.0	38.4	44.0	72.9	54.9	45.0	41.0	47.5
GPT-4o	21.4	22.1	30.5	38.6	36.8	29.0	34.7	27.9	30.2
Claude-3.7-Sonnet	29.0	24.6	32.8	35.5	46.5	35.7	38.7	26.1	33.3
Kimi-Latest	21.4	17.4	19.8	29.1	41.0	27.7	25.4	19.9	24.4
<i>Open-source & Thinking</i>									
QVQ-72B-Preview	37.4	27.1	28.8	35.8	57.6	41.6	33.5	29.1	35.2
Virgo-72B	30.4	22.9	26.1	36.2	47.2	37.7	32.6	24.4	31.8
VL-Rethinker-72B	33.6	28.4	31.4	37.2	59.7	39.0	36.0	31.9	35.8
VL-Rethinker-7B	24.7	17.7	23.5	39.4	42.4	34.4	29.9	22.9	29.3
MM-Eureka-Qwen-32B	23.0	25.7	25.6	36.2	50.7	32.9	30.5	28.1	30.6
MM-Eureka-Qwen-7B	27.1	19.3	22.3	31.9	50.0	32.7	28.7	22.6	28.2
Vision-R1-7B	18.2	18.0	17.9	34.4	36.1	27.4	26.3	18.1	24.0
R1-Onevision-7B-RL	19.5	12.2	20.0	31.6	27.1	27.7	24.8	14.6	22.5
Kimi-VL-A3B-T	28.7	16.0	19.5	32.3	35.4	33.3	25.1	18.1	25.9
<i>Open-source & Chat</i>									
Qwen2.5-VL-72B	31.7	25.1	27.2	37.9	53.5	39.0	32.3	29.9	34.1
Qwen2.5-VL-32B	32.2	26.8	24.4	39.0	52.1	40.5	27.5	29.6	33.2
Qwen2.5-VL-7B	22.2	18.2	21.9	35.1	36.1	31.4	27.5	20.9	26.8
InternVL3-78B	26.0	24.0	26.5	41.8	50.0	35.1	33.8	27.1	32.1
InternVL3-38B	23.0	18.5	23.0	38.3	41.7	33.5	29.0	22.1	28.4
InternVL3-8B	19.5	19.6	22.6	31.6	41.0	28.1	29.9	21.4	26.4
Molmo-72B	12.5	11.9	14.7	28.7	28.5	23.1	18.4	14.3	18.9
Molmo-7B-D	11.7	8.6	8.1	27.3	23.6	20.7	10.9	11.1	14.7
LLaVA-OV-72B	17.1	18.0	23.9	32.3	38.9	27.4	30.5	19.9	25.8
Kimi-VL-A3B	18.7	11.9	21.4	34.0	27.8	25.9	26.3	17.1	23.1

Prominent bias in logical reasoning performance within MLLMs. In almost all cases, models exhibit dominant deductive reasoning performance, while abductive reasoning is considerably weaker. Closed-source models demonstrate an average deductive advantage of 5.38 over abductive reasoning, which widens to 9.81 among open-source models, making abductive reasoning a significant bottleneck in reasoning performance. Deductive reasoning maintains a high proportion in the training corpus due to its widespread distribution. Abductive reasoning processes usually involve larger exploration spaces and richer assumptions, hypotheses, and reflections, making their data challenging to scale. However, in the real world, non-deductive reasoning plays a crucial role in many scenarios such as scientific discoveries. These findings highlight the necessity for researchers to develop a more comprehensive understanding of models’ logical reasoning abilities to facilitate their application in real-world scenarios.

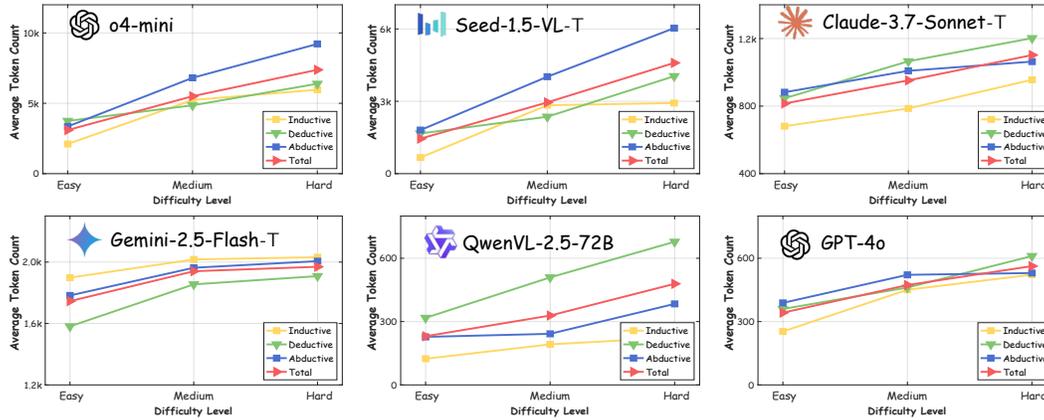


Figure 6: Comparison of Difficulty Level and Average Token Count on MME-Reasoning.

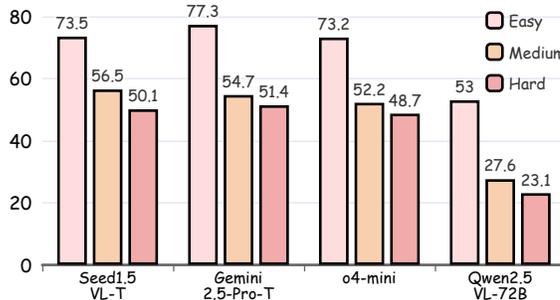


Figure 7: Results within different difficulty levels.

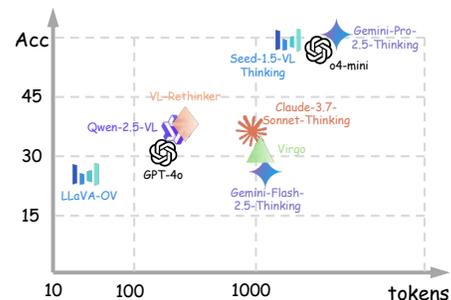


Figure 8: Response tokens vs. Performance.

Comparison with Human Performance. To evaluate expert-level performance on MME-Reasoning, we further report human performance on the mini-set of MME-Reasoning. As shown in Tab. 10, the human expert achieved an overall score of 83.4—significantly outperforming the best-performing thinking model. Looking deeper into the reasoning types, the human expert scored 85.8, 76.9, and 85.6 on deductive, inductive, and abductive reasoning, respectively, all of which are notably higher than the scores of the best-performing model. Moreover, the human expert demonstrated a particularly strong ability in abductive reasoning, with performance comparable to that in deductive reasoning—which is the key focus in current multi-modal reasoning research. This strength aligns with a few top-performing models, but stands in contrast to most baseline models, which show clear weaknesses in abductive reasoning. These results highlight the significant gap that still exists between current thinking & chat models and human-level performance in comprehensive multimodal reasoning evaluation.

Limited performance in open-ended reasoning scenarios. Models generally demonstrate relative advantages in Casual Chain Analysis but perform poorly on tasks involving Plan & Exploration. This may benefit from the autoregressive paradigm, continuously aiding models in learning causal dependencies within input sequences. However, it also highlights a critical shortcoming: current models struggle with planning and exploration in open-ended problem-solving spaces. To advance models in solving difficult practical problems, it is critical to innovate learning paradigms and strategies suitable for open scenarios.

Comprehensive improvement of reasoning ability requires more attention. In most cases, “thinking models” significantly outperform their base version. For example, Seed1.5-VL-T outperformed Seed1.5-VL by 12.4. However, we observed that methods adopting rule-based RL which activating the “thinking mode” of foundational models do not consistently outperform their base models on MME-Reasoning. Most models at the 7B scale experienced performance degradation. This may be due to the training method, data, since existing works often concentrate on a small set of reasoning (*i.e.*, Math). We hope that more researchers can pay attention to a more comprehensive type of reasoning instead of only math.

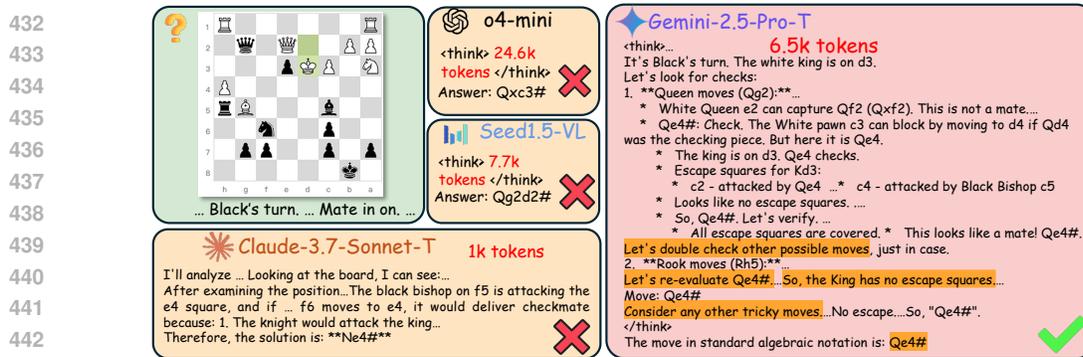


Figure 9: Case study of a Mate-in-one problem.

Table 4: Human and LLM-based evaluation.

	Human Group 1	Human Group 2	Human Group 3	GPT-4o
MathVista	55.7%	65.3%	52.2%	60.7%
MME-Reasoning	96.1%	97.3%	95.8%	94.4%

Table 5: Knowledge concepts.

Benchmark	Question numbers
MMM	3.9
MME-Reasoning	1.4

4.4 FINE GRAINED ANALYSIS

Does increasing the length of the reasoning process help? To investigate whether increased output length consistently leads to improved accuracy, we selected 10 representative models, including Chat Models (e.g., GPT-4o) and Thinking Models (e.g., o4-mini). In Fig. 8, we present the semi-log plot of average token count (ATC) versus accuracy. The overall trend reveals that models with longer outputs tend to achieve higher scores, indicating the effectiveness of extending the reasoning process to enhance reasoning performance. However, we observe that the profit brought by the increase of model output tokens is diminishing. Compared to Thinking Models, Chat Models demonstrate higher token efficiency. These findings highlight the computational cost associated with scaling up inference for improved performance. Balancing reasoning efficiency and model effectiveness remains a challenge for future research.

Is the length of the reasoning process strongly correlated with task difficulty? To examine whether models spontaneously allocate more inference budget to more challenging questions, we conducted research on representative Thinking and Chat Models. We first analyzed the accuracy of different models across varying levels of difficulty, as shown in Fig. 7. With increasing difficulty, model performance declines significantly, confirming the validity of MME-Reasoning’s difficulty stratification. Besides, Fig. 6 illustrates the trend of ATC across different reasoning types and difficulty levels. It reveals a consistent pattern: overall, output length increases steadily with rising difficulty. This trend holds across varying output lengths, model categories, and reasoning types. Compared to Chat Models, Thinking Models exhibit a more pronounced increase in ATC as difficulty rises. For instance, the ATC of Seed1.5-VL increases by up to 3k tokens. In contrast, the ATC increase for Qwen2.5-VL and GPT-4o remains within 300 tokens.

Is MME-Reasoning a reasoning-centric benchmark? We conduct experiments involving both human and LLM-based evaluation to show that MME-Reasoning is a reasoning-centric benchmark. **Human Evaluation:** We enlisted six PhD-level human experts, grouped into pairs (3 groups). Each pair assessed the questions from both MathVista and MME-Reasoning benchmarks to classify them based on the need for reasoning. **LLM-Based Evaluation:** We employ GPT-4o to assess the same classification task, further validating the reasoning-centric nature of our benchmark. Tab. 4 presents the proportion of questions requiring reasoning in each benchmark. The results show that the vast majority of questions in MME-Reasoning (over 95%) were classified as requiring reasoning to solve, which further verifies that MME-Reasoning is a reasoning-centric benchmark. Further, to show that our benchmark requires less knowledge, we follow VisualPuzzles Song et al. (2025) to prompt GPT-4o to generate “knowledge concept checklists” for 50 randomly selected questions and manually verify each question. The results are shown in the following table, which further demonstrates that questions in MME-Reasoning need less advanced knowledge.

486 4.5 CASE STUDY

487
488 In Fig. 9, we present an example of abductive reasoning which demands planning and exploration.
489 From this case, several key observations can be identified: **(1)Long reasoning process**: The se-
490 lected models generated over 1k tokens in response, with o4-mini producing up to 24.6k tokens.
491 This demonstrates that MME-Reasoning constitutes a highly challenging benchmark for multimodal
492 reasoning. **(2)Planning in the problem-solving process**: The response includes multiple iterations
493 of “*hypothesis generation (possible movement) – feasibility verification (check escape squares) –*
494 *check*”, indicating that the model spontaneously engages in structured planning and reflection to
495 explore solutions within an open-ended problem-solving spaces. **(3)Repetitive reflection**: We ob-
496 served that the model tends to revisit and reflect on the same reasoning paths multiple times—up
497 to 7 instances in some cases. This behavior may result in significant computational overhead and
498 informational redundancy. Balancing reasoning efficiency with performance remains a critical issue
499 to be addressed.

500 5 CONCLUSION

501
502 We introduce MME-Reasoning, a comprehensive benchmark designed to evaluate MLLMs’ logical
503 reasoning abilities across inductive, deductive, and abductive reasoning types. Through careful data
504 curation and an expanded evaluation protocol, our benchmark provides a holistic assessment of rea-
505 soning capabilities, beyond simple perception or high-level knowledge. Our experiments reveal that
506 existing MLLMs still face significant challenges and exhibit notable performance imbalances across
507 different reasoning types. These findings underscore the need for further research and development
508 to enhance the reasoning abilities of MLLMs, paving the way for more generalizable AI systems.

509 REPRODUCIBILITY STATEMENT

510
511 To facilitate future research and ensure that our findings can be independently replicated, we have
512 documented our methodology in detail. We provide a thorough breakdown of the data annotation
513 process in the Appendix, offering insights into how the ground truth was established. Additionally,
514 all necessary information regarding the experimental configuration is outlined in the experiment
515 section. This includes a detailed specification of the model families used in our evaluation, as well
516 as the step-by-step protocol followed for the human study.

517 ETHICAL STATEMENT

518
519 This study utilizes samples extracted from existing quiz sources exclusively for the purposes of
520 scholarly analysis and algorithmic testing. We emphasize that the included data samples are neither
521 intended to nor capable of serving as a substitute for the original works. Furthermore, our human
522 evaluation studies were conducted with fair compensation and informed consent from all partici-
523 pants.

524 REFERENCES

- 525
526
527
528 OpenAI (2025). Openai o3 and o4-mini system card, 2025. URL [https://openai.com/](https://openai.com/index/o3-o4-mini-system-card/)
529 [index/o3-o4-mini-system-card/](https://openai.com/index/o3-o4-mini-system-card/).
530
531 <https://www.anthropic.com/index/introducing-claude> Anthropic. Claude, 2022. URL [https://](https://www.anthropic.com/index/introducing-claude)
532 www.anthropic.com/index/introducing-claude.
533
534
535 Gilad Baruch, Zhuoyuan Chen, Afshin Dehghan, Tal Dimry, Yuri Feigin, Peter Fu, Thomas Gebauer,
536 Brandon Joffe, Daniel Kurz, Arik Schwartz, et al. Arkitscenes: A diverse real-world dataset for
537 3d indoor scene understanding using mobile rgb-d data. *arXiv preprint arXiv:2111.08897*, 2021.
538
539 Huanqia Cai, Yijun Yang, and Winston Hu. Mm-iq: Benchmarking human-like abstraction and
reasoning in multimodal models. *arXiv preprint arXiv:2502.00698*, 2025.

- 540 Liang Chen, Lei Li, Haozhe Zhao, Yifan Song, and Vinci. R1-v: Reinforcing super generalization
541 ability in vision-language models with less than \$3. [https://github.com/Deep-Agent/
542 R1-v](https://github.com/Deep-Agent/R1-v), 2025. Accessed: 2025-02-02.
- 543
544 Zhenfang Chen, Qinzhong Zhou, Yikang Shen, Yining Hong, Hao Zhang, and Chuang Gan. See,
545 think, confirm: Interactive prompting between vision and language models for knowledge-based
546 visual reasoning. *arXiv preprint arXiv:2301.05226*, 2023.
- 547
548 Yew Ken Chia, Vernon Toh Yan Han, Deepanway Ghosal, Lidong Bing, and Soujanya Poria. Puz-
549 zlevqa: Diagnosing multimodal reasoning challenges of language models with abstract visual
550 patterns. *arXiv preprint arXiv:2403.13315*, 2024.
- 551
552 Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias
553 Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the
554 IEEE conference on computer vision and pattern recognition*, pp. 5828–5839, 2017.
- 555
556 DeepSeek-AI. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning,
557 2025. URL <https://arxiv.org/abs/2501.12948>.
- 558
559 Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Moham-
560 madreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. Molmo and pixmo: Open
561 weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*,
562 2024.
- 563
564 Yihe Deng, Hritik Bansal, Fan Yin, Nanyun Peng, Wei Wang, and Kai-Wei Chang. Openvlthinker:
565 An early exploration to complex vision-language reasoning via iterative self-improvement. *arXiv
566 preprint arXiv:2503.17352*, 2025.
- 567
568 Yifan Du, Zikang Liu, Yifan Li, Wayne Xin Zhao, Yuqi Huo, Bingning Wang, Weipeng Chen, Zheng
569 Liu, Zhongyuan Wang, and Ji-Rong Wen. Virgo: A preliminary exploration on reproducing o1-
570 like mllm. *arXiv preprint arXiv:2501.01904*, 2025.
- 571
572 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
573 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.
574 *ArXiv preprint, abs/2407.21783*, 2024. URL <https://arxiv.org/abs/2407.21783>.
- 575
576 Kaituo Feng, Kaixiong Gong, Bohao Li, Zonghao Guo, Yibing Wang, Tianshuo Peng, Benyou
577 Wang, and Xiangyu Yue. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint
578 arXiv:2503.21776*, 2025.
- 579
580 Chaoyou Fu, Peixian Chen, Yunhang Shen, Yulei Qin, Mengdan Zhang, Xu Lin, Jinrui Yang, Xiawu
581 Zheng, Ke Li, Xing Sun, et al. Mme: A comprehensive evaluation benchmark for multimodal
582 large language models. *arXiv preprint arXiv:2306.13394*, 2023.
- 583
584 Chaoyou Fu, Yuhang Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu
585 Zhou, Yunhang Shen, Mengdan Zhang, et al. Video-mme: The first-ever comprehensive evalua-
586 tion benchmark of multi-modal llms in video analysis. *arXiv preprint arXiv:2405.21075*, 2024.
- 587
588 Howard Gardner. *Frames of mind: The theory of multiple intelligences*. Basic books, 2011.
- 589
590 Gemini, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu
591 Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable
592 multimodal models. *ArXiv preprint, abs/2312.11805*, 2023. URL [https://arxiv.org/
593 abs/2312.11805](https://arxiv.org/abs/2312.11805).
- 594
595 Kaixiong Gong, Kaituo Feng, Bohao Li, Yibing Wang, Mofan Cheng, Shijia Yang, Jiaming Han,
596 Benyou Wang, Yutong Bai, Zhuoran Yang, et al. Av-odyssey bench: Can your multimodal llms
597 really understand audio-visual information? *arXiv preprint arXiv:2412.02611*, 2024.
- 598
599 Dong Guo, Faming Wu, Feida Zhu, Fuxing Leng, Guang Shi, Haobin Chen, Haoqi Fan, Jian Wang,
600 Jianyu Jiang, Jiawei Wang, et al. Seed1.5-vl technical report. *arXiv preprint arXiv:2505.07062*,
601 2025a.

- 594 Meng-Hao Guo, Xuanyu Chu, Qianrui Yang, Zhe-Han Mo, Yiqing Shen, Pei-lin Li, Xinjie Lin,
595 Jinnian Zhang, Xin-Sheng Chen, Yi Zhang, et al. Rbench-v: A primary assessment for visual
596 reasoning models with multi-modal outputs. *arXiv preprint arXiv:2505.16770*, 2025b.
- 597
- 598 Yunzhuo Hao, Jiawei Gu, Huichen Will Wang, Linjie Li, Zhengyuan Yang, Lijuan Wang, and
599 Yu Cheng. Can mllms reason in multimodality? emma: An enhanced multimodal reasoning
600 benchmark. *arXiv preprint arXiv:2501.05444*, 2025.
- 601
- 602 Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu,
603 Xu Han, Yujie Huang, Yuxiang Zhang, et al. Olympiadbench: A challenging benchmark for
604 promoting agi with olympiad-level bilingual multimodal scientific problems. *arXiv preprint
arXiv:2402.14008*, 2024.
- 605
- 606 Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith,
607 and Ranjay Krishna. Visual sketchpad: Sketching as a visual chain of thought for multimodal
608 language models. *arXiv preprint arXiv:2406.09403*, 2024.
- 609
- 610 Wenxuan Huang, Bohan Jia, Zijie Zhai, Shaosheng Cao, Zheyu Ye, Fei Zhao, Zhe Xu, Yao Hu, and
611 Shaohui Lin. Vision-r1: Incentivizing reasoning capability in multimodal large language models.
arXiv preprint arXiv:2503.06749, 2025.
- 612
- 613 Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec
614 Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. Openai o1 system card. *arXiv
preprint arXiv:2412.16720*, 2024.
- 615
- 616 Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanwei Li, Yu Qi, Xinyan Chen, Liuhui Wang, Jianhan
617 Jin, Claire Guo, Shen Yan, Bo Zhang, Chaoyou Fu, Peng Gao, and Hongsheng Li. Mme-cot:
618 Benchmarking chain-of-thought in large multimodal models for reasoning quality, robustness,
619 and efficiency, 2025. URL <https://arxiv.org/abs/2502.09621>.
- 620
- 621 Yilei Jiang, Yingshui Tan, and Xiangyu Yue. Rapguard: Safeguarding multimodal large language
622 models via rationale-aware defensive prompting, 2024. URL [https://arxiv.org/abs/
2412.18826](https://arxiv.org/abs/2412.18826).
- 623
- 624 Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and
625 Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual
626 reasoning, 2016. URL <https://arxiv.org/abs/1612.06890>.
- 627
- 628 Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei
629 Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *arXiv preprint
arXiv:2408.03326*, 2024.
- 630
- 631 Hanmeng Liu, Zhizhang Fu, Mengru Ding, Ruoxi Ning, Chaoli Zhang, Xiaozhang Liu, and Yue
632 Zhang. Logical reasoning in large language models: A survey. *arXiv preprint arXiv:2502.09100*,
633 2025a.
- 634
- 635 Ziyu Liu, Zeyi Sun, Yuhang Zang, Xiaoyi Dong, Yuhang Cao, Haodong Duan, Dahua Lin, and Jiaqi
636 Wang. Visual-rft: Visual reinforcement fine-tuning. *arXiv preprint arXiv:2503.01785*, 2025b.
- 637
- 638 Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-
639 Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of
640 foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.
- 641
- 642 Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-
643 Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning
644 of foundation models in visual contexts. In *The Twelfth International Conference on Learning
645 Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL
<https://openreview.net/forum?id=KUNzEQMWU7>.
- 646
- 647 Yiting Lu, Jiakang Yuan, Zhen Li, Shitian Zhao, Qi Qin, Xinyue Li, Le Zhuo, Licheng Wen,
Dongyang Liu, Yuwen Cao, et al. Omnicaptioner: One captioner to rule them all. *arXiv preprint
arXiv:2504.07089*, 2025.

- 648 Fanqing Meng, Lingxiao Du, Zongkai Liu, Zhixiang Zhou, Quanfeng Lu, Daocheng Fu, Botian
649 Shi, Wenhai Wang, Junjun He, Kaipeng Zhang, Ping Luo, Yu Qiao, Qiaosheng Zhang, and Wenqi
650 Shao. Mm-eureka: Exploring visual aha moment with rule-based large-scale reinforcement learn-
651 ing, 2025. URL <https://github.com/ModalMinds/MM-EUREKA>.
- 652 OpenAI. Hello gpt4-o. <https://openai.com/index/hello-gpt-4o/>, 2024. URL <https://openai.com/index/hello-gpt-4o/>.
- 655 Charles Sanders Peirce. *Illustrations of the Logic of Science*. Open Court, 2014.
- 656 Tianshuo Peng, Mingsheng Li, Hongbin Zhou, Renqiu Xia, Renrui Zhang, Lei Bai, Song Mao,
657 Bin Wang, Conghui He, Aojun Zhou, et al. Chimera: Improving generalist model with domain-
658 specific experts. *arXiv preprint arXiv:2412.05983*, 2024.
- 659 Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang,
660 Xingzhong Xu, Xin Geng, and Xu Yang. Lmm-r1: Empowering 3b llms with strong reasoning
661 abilities through two-stage rule-based rl. *arXiv preprint arXiv:2503.07536*, 2025a.
- 662 Yingzhe Peng, Gongrui Zhang, Miaosen Zhang, Zhiyuan You, Jie Liu, Qipeng Zhu, Kai Yang,
663 Xingzhong Xu, Xin Geng, and Xu Yang. Lmm-r1: Empowering 3b llms with strong reasoning
664 abilities through two-stage rule-based rl. *arXiv preprint arXiv:2503.07536*, 2025b.
- 665 Qwen Team. Qwen2.5-vl, January 2025a. URL <https://qwenlm.github.io/blog/qwen2.5-vl/>.
- 666 Qwen Team. Qwq-32b: Embracing the power of reinforcement learning, March 2025b. URL
667 <https://qwenlm.github.io/blog/qwq-32b/>.
- 668 David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien
669 Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof qa bench-
670 mark, 2023. URL <https://arxiv.org/abs/2311.12022>.
- 671 Haozhan Shen, Peng Liu, Jingcheng Li, Chunxin Fang, Yibo Ma, Jiajia Liao, Qiaoli Shen, Zilun
672 Zhang, Kangjia Zhao, Qianqian Zhang, Ruochen Xu, and Tiancheng Zhao. Vlm-r1: A stable and
673 generalizable r1-style large vision-language model, 2025. URL [https://arxiv.org/abs/](https://arxiv.org/abs/2504.07615)
674 [2504.07615](https://arxiv.org/abs/2504.07615).
- 675 Yueqi Song, Tianyue Ou, Yibo Kong, Zecheng Li, Graham Neubig, and Xiang Yue. Visualpuz-
676 zles: Decoupling multimodal reasoning evaluation from domain knowledge. *arXiv preprint*
677 *arXiv:2504.10342*, 2025.
- 678 Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun
679 Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with
680 llms. *arXiv preprint arXiv:2501.12599*, 2025a.
- 681 Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen,
682 Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-vl technical report. *arXiv preprint*
683 *arXiv:2504.07491*, 2025b.
- 684 Qwen Team. Qvq: To see the world with wisdom, December 2024. URL [https://qwenlm.](https://qwenlm.github.io/blog/qvq-72b-preview/)
685 [github.io/blog/qvq-72b-preview/](https://qwenlm.github.io/blog/qvq-72b-preview/).
- 686 Omkar Thawakar, Dinura Dissanayake, Ketan More, Ritesh Thawkar, Ahmed Heakl, Noor Ahsan,
687 Yuhao Li, Mohammed Zumri, Jean Lahoud, Rao Muhammad Anwer, et al. Llamav-o1: Rethink-
688 ing step-by-step visual reasoning in llms. *arXiv preprint arXiv:2501.06186*, 2025.
- 689 Haozhe Wang, Chao Qu, Zuming Huang, Wei Chu, Fangzhen Lin, and Wenhui Chen. Vl-
690 rethinker: Incentivizing self-reflection of vision-language models with reinforcement learning.
691 *arXiv preprint arXiv:2504.08837*, 2025.
- 692 Ke Wang, Juntong Pan, Weikang Shi, Zimu Lu, Houxing Ren, Aojun Zhou, Mingjie Zhan, and Hong-
693 sheng Li. Measuring multimodal mathematical reasoning with math-vision dataset. *Advances in*
694 *Neural Information Processing Systems*, 37:95095–95169, 2024a.

- 702 Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu,
703 Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model’s perception of the
704 world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.
- 705
- 706 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny
707 Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in*
708 *neural information processing systems*, 35:24824–24837, 2022.
- 709
- 710 Renqiu Xia, Mingsheng Li, Hancheng Ye, Wenjie Wu, Hongbin Zhou, Jiakang Yuan, Tianshuo
711 Peng, Xinyu Cai, Xiangchao Yan, Bin Wang, et al. Geox: Geometric problem solving through
712 unified formalized vision-language pre-training. *arXiv preprint arXiv:2412.11863*, 2024.
- 713
- 714 Weiye Xu, Jiahao Wang, Weiyun Wang, Zhe Chen, Wengang Zhou, Aijun Yang, Lewei Lu,
715 Houqiang Li, Xiaohua Wang, Xizhou Zhu, et al. Visulogic: A benchmark for evaluating visual
716 reasoning in multi-modal large language models. *arXiv preprint arXiv:2504.15279*, 2025.
- 717
- 718 Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in
719 space: How multimodal large language models see, remember, and recall spaces. *arXiv preprint*
arXiv:2412.14171, 2024.
- 720
- 721 Yi Yang, Xiaoxuan He, Hongkun Pan, Xiyan Jiang, Yan Deng, Xingtao Yang, Haoyu Lu, Dacheng
722 Yin, Fengyun Rao, Minfeng Zhu, et al. R1-onevision: Advancing generalized multimodal rea-
723 soning through cross-modal formalization. *arXiv preprint arXiv:2503.10615*, 2025.
- 724
- 725 Fanglong Yao, Changyuan Tian, Jintao Liu, Zequn Zhang, Qing Liu, Li Jin, Shuchao Li, Xiaoyu
726 Li, and Xian Sun. Thinking like an expert: Multimodal hypergraph-of-thought (hot) reasoning to
727 boost foundation modals. *arXiv preprint arXiv:2308.06207*, 2023.
- 728
- 729 Huanjin Yao, Jiaying Huang, Wenhao Wu, Jingyi Zhang, Yibo Wang, Shunyu Liu, Yingjie Wang,
730 Yuxin Song, Haocheng Feng, Li Shen, et al. Mulberry: Empowering mllm with o1-like reasoning
731 and reflection via collective monte carlo tree search. *arXiv preprint arXiv:2412.18319*, 2024.
- 732
- 733 Peng Yingzhe, Zhang Gongrui, Zhang Miaosen, You Zhiyuan, Liu Jie, Zhu Qipeng, Yang Kai,
734 Xu Xingzhong, Geng Xin, and Yang Xu. Lmm-r1: Empowering 3b llms with strong reasoning
735 abilities through two-stage rule-based rl, 2025.
- 736
- 737 Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens,
738 Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun,
739 Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, Huan Sun, Yu Su, and
740 Wenhao Chen. Mmmu: A massive multi-discipline multimodal understanding and reasoning
741 benchmark for expert agi. In *Proceedings of CVPR*, 2024a.
- 742
- 743 Xiang Yue, Tianyu Zheng, Yuansheng Ni, Yubo Wang, Kai Zhang, Shengbang Tong, Yuxuan Sun,
744 Botao Yu, Ge Zhang, Huan Sun, et al. Mmmu-pro: A more robust multi-discipline multimodal
745 understanding benchmark. *arXiv preprint arXiv:2409.02813*, 2024b.
- 746
- 747 Jingyi Zhang, Jiaying Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng
748 Tao. R1-vl: Learning to reason with multimodal large language models via step-wise group
749 relative policy optimization. *arXiv preprint arXiv:2503.12937*, 2025a.
- 750
- 751 Renrui Zhang, Dongzhi Jiang, Yichi Zhang, Haokun Lin, Ziyu Guo, Pengshuo Qiu, Aojun Zhou,
752 Pan Lu, Kai-Wei Chang, Yu Qiao, et al. Mathverse: Does your multi-modal llm truly see the
753 diagrams in visual math problems? In *European Conference on Computer Vision*, pp. 169–186.
754 Springer, 2024.
- 755
- 756 Zeyu Zhang, Zijian Chen, Zicheng Zhang, Yuze Sun, Yuan Tian, Ziheng Jia, Chunyi Li, Xiaohong
757 Liu, Xiongkuo Min, and Guangtao Zhai. Puzzlebench: A fully dynamic evaluation framework
758 for large multimodal models on puzzle solving. *arXiv preprint arXiv:2504.10885*, 2025b.
- 759
- 760 Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal
761 chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023.

756 Changmeng Zheng, Dayong Liang, Wengyu Zhang, Xiao-Yong Wei, Tat-Seng Chua, and Qing Li. A
757 picture is worth a graph: A blueprint debate paradigm for multimodal reasoning. In *Proceedings*
758 *of the 32nd ACM International Conference on Multimedia*, pp. 419–428, 2024.

759
760 Hengguang Zhou, Xirui Li, Ruochen Wang, Minhao Cheng, Tianyi Zhou, and Cho-Jui Hsieh. R1-
761 zero’s ”aha moment” in visual reasoning on a 2b non-sft model, 2025. URL <https://arxiv.org/abs/2503.05132>.

762
763 Luowei Zhou, Chenliang Xu, and Jason Corso. Towards automatic learning of procedures from web
764 instructional videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32,
765 2018.

766
767 Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao
768 Tian, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for
769 open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809

810	APPENDIX	
811		
812		
813		
814	A More Detailed Comparison Between MME-Reasoning and Other Benchmarks	16
815		
816	B More Experimental Results	17
817	B.1 Full Results on MME-Reasoning	17
818	B.2 Full Results on Mini-set of MME-Reasoning	17
819	B.3 Further Analysis of Human Performance	19
820	B.4 Results on Different Question Types	20
821	B.5 Results with Test-Time Compute Scaling	20
822	B.6 Results with CoT Prompt	20
823	B.7 Token Usage of Thinking Models	21
824	B.8 Results of Captioner & LLMs	22
825	B.9 Response Length of Different Benchmarks	23
826		
827		
828		
829		
830		
831	C Details of Annotation	23
832	C.1 Overall Annotation Process	23
833	C.2 Difficult Annotation	23
834	C.3 Reasoning Type Annotation	24
835	C.4 Capability Annotation	24
836		
837		
838		
839	D Details of Implementation	24
840	D.1 Implementation Details	24
841		
842	E Details of Evaluation	25
843	E.1 Prompts for Answer Extraction	25
844		
845		
846	F Examples of MME-Reasoning	25
847		
848	G Limitation	25
849		
850		
851	A MORE DETAILED COMPARISON BETWEEN MME-REASONING AND	
852	OTHER BENCHMARKS	
853		

In this section, we provide a more detailed comparison between MME-Reasoning and other benchmarks as shown in the Tab. 6. We further provide here a mapping between the sub-reasoning types in our dataset and the three major reasoning types in Tab. 7.

The difference can be summarized into the following three aspects:

- **Covering More Reasoning Types and Subtypes:** MME-Reasoning includes a broader range of reasoning types and subtypes. First, based on the three-stage theory of reasoning (premise, rules, and result), we categorize reasoning into three major classes (*i.e.*, inductive, deductive, and abductive), where two can infer the third. While previous reasoning benchmarks typically only cover inductive and deductive reasoning, our benchmark comprehensively incorporates abductive reasoning as well. Furthermore, compared to existing

864 benchmarks, which usually divide the reasoning types into more fine-grained subtypes such
 865 as spatial or mathematical reasoning, MME-Reasoning covers an even wider set of reason-
 866 ing subtypes including spatial reasoning, temporal reasoning, puzzle (quantitative, spatial,
 867 positional, attribute, stylistic), geometry, algebraic, statistical, scientific (physics, biology,
 868 chemistry) (as shown in the table), enabling a more comprehensive evaluation of MLLMs’
 869 reasoning abilities.

- 870 • Diversity of Question Formats: In terms of question types, existing benchmarks usually
 871 focus on MCQ, while MME-Reasoning encompasses three formats: MCQ, free-form, and
 872 rule-based. This allows us to cover a wider variety of question types, such as Sudoku and
 873 Number Bridge, and so on, which are not typically included in previous benchmarks.
- 874 • Variety of Image Types: MME-Reasoning also includes a wider range of image types.
 875 Compared to earlier reasoning benchmarks, we comprehensively cover images from Math
 876 (Geometry, Function), Biology, Chemistry, Physics, Charts, Puzzles (Quantitative, Spatial,
 877 Positional, Attribute, Stylistic, Chess and Card, Sudoku), Multi-view images, Temporal
 878 sequences, and others (such as chess). This extensive coverage ensures that our benchmark
 879 can more thoroughly assess the reasoning capabilities of MLLMs.

880 Table 6: More detailed comparison between MME-Reasoning and other benchmarks.
 881

882 Benchmark	883 Size	884 Question Types		885 Answer Types	886 Reasoning Types	887 Image Types	888 Sub reasoning / Task Types
889 MathVista	6.1k	MCQ,	Free-	Text	Inductive, De-	Math (Geometry, Function), Chart, Table, Puzzle, Others (Synthetic scene, Abstract Scene)	Geometry, Algebraic, Statis- tical, Scientific, Puzzles
890 MathVerse	11.5K	MCQ,	Free-	Text	Deductive	Math (Geometry, Function)	Geometry, Algebraic
891 LogicVista	0.4K	MCQ		Image	Inductive, De-	Text-rich image, Table, Chart, Physics, Puzzles, Others (3D shape)	Spatial, Scientific, Puzzles
892 VisualPuzzles	1.2K	MCQ		Image+Text	Inductive	Puzzles (Quantitative, Spa- tial)	Puzzles (Quantitative, Spa- tial)
893 VisuLogic	1K	MCQ		Image	Inductive	Puzzles (Quantitative, Spa- tial, Positional, Attribute, Stylistic)	Puzzles (Quantitative, Spa- tial, Positional, Attribute, Stylistic, Others)
894 Rbench-V	803	MCQ,	Free-	Image+Text	Inductive, De-	Math, Physics, Counting, Puzzles, Games	Geometry, Physics, Count- ing, Puzzle, Games
895 MM-IQ	2.7K	MCQ		Image+Text	Inductive	Puzzles	Puzzles
896 MME-Reasoning	1.2K	MCQ, Free- form, Rule- based		Image+Text	Inductive, De- ductive, Ab- ductive	Math (Geometry, func- tion), Biology, Chemistry, Physics, Chart, Puzzles (Quantitative, Spatial, Posi- tional, Attribute, Stylistic, chess and card, sudoku), Multi-view image, Tempo- ral sequence, Others (chess)	Spatial, Temporal, Puzzle (Quantitative, Spatial, Posi- tional, Attribute, Stylistic, Others), Geometry, Alge- braic, Statistical, Scientific (Physics, Chemistry, Biol- ogy)

904
905
906 **B MORE EXPERIMENTAL RESULTS**

907
908 **B.1 FULL RESULTS ON MME-REASONING**

909 We present the performance of more baselines on MME-Reasoning in Tab 8, including OpenVL-
 910 Thinker (Deng et al., 2025), LMM-R1-MGT-PerceReason (Peng et al., 2025a), Mulberry (Yao et al.,
 911 2024), LlamaV-o1 (Thawakar et al., 2025) and Qwen2-VL series (Wang et al., 2024b).
 912

913
914 **B.2 FULL RESULTS ON MINI-SET OF MME-REASONING**

915 We randomly sampled 25% of the questions and conducted manual review to ensure that the diversity
 916 of image types was maintained. These sampled questions were then used to to construct the Mini-
 917 set. We also analyzed the question distributions of both the Mini-set and the Full-set to ensure the
 sampled questions retained the same distribution. The statistical results are presented in Tab 9.

Table 7: The mapping between reasoning sub-types and major reasoning types.

Reasoning Sub-type	Deductive	Inductive	Abductive
Spatial Reasoning	✓		✓
Temporal Reasoning			✓
Geometry Reasoning	✓		
Algebraic Reasoning	✓		
Statistical Reasoning	✓		
Scientific Reasoning-Physic	✓		✓
Scientific Reasoning-Biology	✓		✓
Scientific Reasoning-Chemistry	✓		✓
Puzzle-Quantitative Reasoning	✓	✓	
Puzzle-Spatial Reasoning	✓	✓	✓
Puzzle-Positional Reasoning		✓	
Puzzle-Attribute Reasoning		✓	
Puzzle-Stylistic Reasoning		✓	
Puzzle-Others Nikoli-style Puzzles and Chess			✓

Table 8: Performance comparison of state-of-the-art MLLMs on MME-Reasoning. The top three are highlighted in blue. “T” represents “Thinking”.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P&E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
<i>Close-source & Thinking</i>									
Gemini-2.5-Pro-T	68.0	64.4	53.7	52.1	90.3	64.0	51.7	62.8	60.2
Seed1.5-VL-T	67.2	62.7	56.0	47.2	82.6	64.5	52.3	60.8	59.9
o4-mini	63.1	58.3	57.2	50.4	59.0	60.6	51.4	59.0	57.5
Claude-4-Sonnet-T	33.3	35.9	33.0	36.2	47.9	39.4	32.0	35.7	36.1
Claude-3.7-Sonnet-T	30.4	27.6	32.3	38.3	46.5	34.6	36.2	31.7	34.1
Gemini-2.5-Flash-T	19.8	21.3	20.9	33.0	38.9	28.1	22.1	24.6	25.2
<i>Close-source & Chat</i>									
Seed1.5-VL	<u>52.0</u>	<u>42.0</u>	<u>38.4</u>	<u>44.0</u>	<u>72.9</u>	<u>54.9</u>	<u>45.0</u>	<u>41.0</u>	<u>47.5</u>
GPT-4o	21.4	22.1	30.5	38.6	36.8	29.0	34.7	27.9	30.2
Claude-3.7-Sonnet	29.0	24.6	32.8	35.5	46.5	35.7	38.7	26.1	33.3
Kimi-Latest	21.4	17.4	19.8	29.1	41.0	27.7	25.4	19.9	24.4
<i>Open-source & Thinking</i>									
QVQ-72B-Preview	<u>37.4</u>	27.1	28.8	35.8	57.6	<u>41.6</u>	33.5	29.1	35.2
Virgo-72B	30.4	22.9	26.1	36.2	47.2	37.7	32.6	24.4	31.8
VL-Rethinker-72B	33.6	<u>28.4</u>	<u>31.4</u>	<u>37.2</u>	<u>59.7</u>	39.0	<u>36.0</u>	<u>31.9</u>	<u>35.8</u>
VL-Rethinker-7B	24.7	17.7	23.5	39.4	42.4	34.4	29.9	22.9	29.3
MM-Eureka-Qwen-32B	23.0	25.7	25.6	36.2	50.7	32.9	30.5	28.1	30.6
MM-Eureka-Qwen-7B	27.1	19.3	22.3	31.9	50.0	32.7	28.7	22.6	28.2
R1-VL-7B	16.3	11.6	17.7	30.9	26.4	25.3	21.8	15.8	21.1
Vision-R1-7B	18.2	18.0	17.9	34.4	36.1	27.4	26.3	18.1	24.0
R1-Onevision-7B-RL	19.5	12.2	20.0	31.6	27.1	27.7	24.8	14.6	22.5
Kimi-VL-A3B-T	28.7	16.0	19.5	32.3	35.4	33.3	25.1	18.1	25.9
OpenVLThinker-7B	19.8	14.6	19.3	35.8	34.7	30.7	24.8	17.3	24.6
LMM-R1-MGT-PerceReason	22.2	16.0	23.7	37.9	34.0	30.3	32.3	20.1	27.4
Mulberry	14.6	13.3	18.8	33.7	31.3	23.8	25.4	17.6	22.1
LlamaV-o1	14.9	7.7	16.5	28.0	25.0	22.4	21.5	12.3	18.8
<i>Open-source & Chat</i>									
Qwen2.5-VL-72B	<u>31.7</u>	25.1	<u>27.2</u>	37.9	<u>53.5</u>	39.0	32.3	<u>29.9</u>	<u>34.1</u>
Qwen2.5-VL-32B	32.2	<u>26.8</u>	24.4	39.0	52.1	<u>40.5</u>	27.5	29.6	33.2
Qwen2.5-VL-7B	22.2	18.2	21.9	35.1	36.1	31.4	27.5	20.9	26.8
Qwen2.5-VL-3B	17.6	15.5	19.0	39.7	32.6	28.5	27.5	19.6	25.6
Qwen2-VL-72B	19.2	19.3	24.9	36.2	44.4	28.8	32.3	22.1	27.5
Qwen2-VL-7B	15.7	12.4	19.8	37.9	30.5	25.5	25.7	19.7	23.4
Qwen2-VL-2B	13.0	8.1	19.3	31.6	19.4	22.7	25.7	11.8	19.9
InternVL3-78B	26.0	24.0	26.5	<u>41.8</u>	50.0	35.1	<u>33.8</u>	27.1	32.1
InternVL3-38B	23.0	18.5	23.0	38.3	41.7	33.5	29.0	22.1	28.4
InternVL3-8B	19.5	19.6	22.6	31.6	41.0	28.1	29.9	21.4	26.4
Molmo-72B	12.5	11.9	14.7	28.7	28.5	23.1	18.4	14.3	18.9
Molmo-7B-D	11.7	8.6	8.1	27.3	23.6	20.7	10.9	11.1	14.7
Molmo-7B-O	8.1	5.5	11.6	22.7	15.3	16.6	16.0	7.5	13.4
LLaVA-OV-72B	17.1	18.0	23.9	32.3	38.9	27.4	30.5	19.9	25.8
Kimi-VL-A3B	18.7	11.9	21.4	34.0	27.8	25.9	26.3	17.1	23.1

Table 9: Comparison of statistics between full and mini-set of MME-Reasoning.

Split	Reasoning Type			Question Type			Difficulty Level		
	DED.	IND.	ABD.	Open	MCQ	Rule.	Easy	Medium	Hard
Mini	39.7%	25.8%	34.4%	32.4%	58.3%	9.3%	31.8%	39.4%	28.8%
Full	38.6%	27.9%	33.5%	31.6%	58.5%	9.9%	30.8%	39.3%	29.9%

Table 10: Performance comparison of state-of-the-art MLLMs on mini set of MME-Reasoning. The top three are highlighted in blue. “T” represents “Thinking”.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P&E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
<i>Human Performance</i>									
Human Expert	75.0	84.4	84.9	80.3	88.1	85.8	76.9	85.6	83.4
<i>Close-source & Thinking</i>									
Gemini-2.5-Pro-T	66.0	63.5	58.5	49.3	85.7	60.8	55.1	65.4	60.9
Seed1.5-VL-T	68.0	67.7	58.5	49.3	83.3	67.5	48.7	67.3	62.6
o1-mini	64.0	58.3	56.6	45.1	54.8	57.5	51.3	60.6	57.0
o1	50.0	38.5	41.5	43.7	52.4	50.8	42.3	42.3	45.7
Claude-4-Sonnet-T	33.0	30.2	35.8	39.4	50.0	42.5	37.2	33.7	38.1
Claude-3.7-Sonnet-T	30.0	17.7	36.8	38.0	38.1	31.7	42.3	27.9	33.1
Gemini-2.5-Flash-T	18.0	16.7	15.1	39.4	33.3	27.5	19.2	26.0	24.8
<i>Close-source & Chat</i>									
Seed1.5-VL	50.0	42.7	34.9	40.8	69.0	57.5	39.7	39.4	46.7
GPT-4o	20.0	24.0	24.5	40.8	33.3	31.7	28.2	27.9	29.5
Claude-3.7-Sonnet	27.0	22.9	34.0	31.0	42.9	31.7	38.5	27.9	32.1
Kimi-Latest	22.0	17.7	17.9	29.6	33.3	30.8	23.1	19.2	24.8
<i>Open-source & Thinking</i>									
QVQ-72B-Preview	36.0	24.0	34.0	33.8	47.6	38.3	37.2	29.8	35.1
Virgo-72B	28.0	18.8	27.4	43.7	38.1	37.5	41.0	21.2	32.8
VL-Rethinker-72B	23.0	25.0	29.2	39.4	42.9	34.2	32.1	31.7	32.8
VL-Rethinker-7B	23.0	16.7	21.7	47.9	40.5	35.8	28.2	26.0	30.5
MM-Eureka-Qwen-32B	23.0	20.8	26.4	38.0	38.1	32.5	34.6	25.0	30.5
MM-Eureka-Qwen-7B	28.0	17.7	21.7	32.4	50.0	32.5	32.1	22.1	28.8
R1-VL-7B	10.0	10.4	16.0	35.2	16.7	23.3	19.2	16.3	19.9
Vision-R1-7B	14.0	12.5	18.9	39.4	31.0	26.7	29.5	16.3	23.8
R1-Onevision-7B-RL	15.0	10.4	22.6	35.2	19.0	22.5	30.8	16.3	22.5
Kimi-VL-A3B-T	30.0	9.4	19.8	26.8	31.0	28.3	26.9	16.3	23.8
OpenVLThinker-7B	14.0	14.6	14.2	33.8	28.6	29.2	16.7	16.3	21.5
LMM-R1-MGT-PerceReason	27.0	14.6	23.6	38.0	33.3	35.8	33.3	18.3	29.1
Mulberry	19.0	15.6	18.9	33.8	33.3	28.3	23.1	18.3	23.5
LlamaV-o1	15.0	8.3	17.9	31.0	26.2	23.3	23.1	15.4	20.5
<i>Open-source & Chat</i>									
Qwen2.5-VL-72B	31.0	19.8	25.5	38.0	42.9	39.2	32.1	26.0	32.8
Qwen2.5-VL-32B	31.0	28.1	28.3	40.8	45.2	41.7	34.6	27.9	35.1
Qwen2.5-VL-7B	19.0	16.7	24.5	38.0	33.3	32.5	30.8	21.2	28.1
Qwen2.5-VL-3B	21.0	14.6	21.2	39.4	31.0	30.0	30.8	21.2	27.2
Qwen2-VL-72B	20.0	19.8	28.3	38.0	38.1	34.2	39.7	19.2	30.5
Qwen2-VL-7B	16.0	9.4	25.5	33.8	26.2	22.5	34.6	16.3	23.5
Qwen2-VL-2B	12.0	9.4	17.9	29.6	19.0	23.3	23.1	11.5	19.2
InternVL3-78B	25.0	22.9	33.0	42.3	40.5	36.7	43.6	24.0	34.1
InternVL3-38B	19.0	19.8	26.4	36.6	38.1	31.7	33.3	23.1	29.1
InternVL3-8B	19.0	20.8	29.2	23.9	35.7	26.7	35.9	21.2	27.2
Molmo-72B	11.0	13.5	16.0	35.2	31.0	26.7	21.8	17.3	22.2
Molmo-7B-D	12.0	8.3	12.3	28.2	16.7	22.5	15.4	9.6	16.2
Molmo-7B-O	7.0	4.2	11.3	25.4	14.3	19.2	17.9	4.8	13.9
LLaVA-OV-72B	13.0	19.8	25.5	23.9	35.7	25.0	30.8	17.3	23.8
Kimi-VL-A3B	18.0	8.3	18.9	29.6	9.5	23.3	23.1	11.5	19.2

We provide the performance of all baseline models on the Mini-set in Tab. 10. All baseline models achieved similar performance on both the Full-set and the Mini-set, further demonstrating the consistency of Mini-set and the comparability of model performance across different splits.

B.3 FURTHER ANALYSIS OF HUMAN PERFORMANCE

In our manuscript, we report PhD-level performance on MME-Reasoning in Tab. 3. For a further assessment of performance across different backgrounds, five college-level students with varied

Table 11: Performance comparison of chat models with or w/o MCTS.

	Deductive	Inductive	Abductive	Average
Human Expert 1	85.8	76.9	85.6	83.4
Human Expert 2	90.1	77.5	86.1	84.6
Human Expert 3	82.1	71.5	86.1	79.9
Human Expert 4	75.5	76.8	83.8	78.7
Human Expert 5	80.8	75.2	85.4	80.5

backgrounds were recruited for testing. The results are presented in Tab. 11. The results show that while there is some performance variability among human participants, likely due to their diverse backgrounds, they consistently outperform the current models. This shows that there is still a performance gap between existing SoTA models and humans. Besides, we observed that the performance variability among human participants was more pronounced on deductive and inductive tasks compared to abductive ones. We attribute this to the fact that performance on puzzles and mathematical reasoning is often influenced by a participant’s specific academic background and expertise. In contrast, abductive reasoning relies on more universal human abilities, such as the capacity for trial, critical thinking, and reflection. Compared to the models, humans have a much stronger ability to learn from trial and error. This results in a more stable and consistently high level of performance on abductive reasoning tasks across participants, regardless of their individual backgrounds.

B.4 RESULTS ON DIFFERENT QUESTION TYPES

We also evaluated the model’s performance across different question types and present the results in Tab. 12.

B.5 RESULTS WITH TEST-TIME COMPUTE SCALING

To evaluate whether the use of Test-Time Compute Scaling (TTS) methods can improve model performance on MME-Reasoning, we take Qwen2.5-VL-7B as an example and use Qwen2.5-VL-32B as the Reward Model. The evaluation is conducted using the Monte Carlo Tree Search (MCTS) algorithm, with the settings: *branch* = 3 and *max-iteration* = 18. The results are shown in Table 13.

Under the MCTS-based setting, the model’s performance dropped noticeably across all reasoning types. We attribute this decline to two main factors: (1) Questions in MME-Reasoning often involve complex parallel reasoning, hypothesis generation, and reflection, rather than simple linear logical progression. These characteristics may not be effectively captured by the Reward Model. (2) The limited capabilities of the Reward Model result in guidance that lacks practical utility.

We leave further exploration of TTS methods for reasoning to future work and hope that MME-Reasoning can serve as a representative benchmark for developing more general and comprehensive TTS algorithms in reasoning tasks.

B.6 RESULTS WITH CoT PROMPT

Chain-of-Thought (CoT) prompting increases output length by encouraging explicit output of the thought process, thereby enhancing reasoning performance. To investigate the impact of CoT on performance in MME-Reasoning, we evaluated the Qwen2.5-VL and InternVL3 series using CoT prompts shown in Tab. 15. The results are presented in Tab. 14.

We observed that the Qwen2.5-VL models naturally tend to generate their reasoning process, so adding a CoT prompt did not significantly increase output length. In contrast, InternVL3 models, under default settings, tend to directly output the final answer, and the CoT prompt substantially increased output length.

In terms of performance, adding the CoT prompt consistently led to performance degradation for the Qwen2.5-VL series. For InternVL3, performance dropped for the 7B model but improved for the larger 38B and 78B models. One possible hypothesis is that for models already inclined to produce long outputs, explicit CoT instructions might introduce noise into the reasoning process. Conversely, for models that tend to answer questions directly, smaller models struggle to produce helpful and

Table 12: Performance across different question types on MME-Reasoning. The top three are highlighted in green. † indicates the model was evaluated on the mini-set. “T” represents “Thinking”.

Model	Choice				Open				Rule
	DED.	IND.	ABD.	ALL	DED.	IND.	ABD.	ALL	ABD.&ALL
<i>Close-source & Thinking</i>									
Gemini-2.5-Pro-T	58.0	49.8	63.6	55.7	75.9	61.5	60.0	66.9	66.1
Seed1.5-VL-T	57.3	54.2	60.2	56.5	78.5	44.2	59.4	65.3	63.5
o4-mini	57.3	48.7	61.9	54.7	67.1	67.3	48.5	58.9	71.3
o1†	46.2	42.4	53.3	46.0	60.0	45.5	36.2	46.9	40.7
Claude-4-Sonnet-T	41.1	33.2	40.7	37.9	36.5	25.0	35.2	34.3	31.3
Claude-3.7-Sonnet-T	38.0	39.7	46.6	40.1	28.5	17.3	31.5	28.3	16.5
Gemini-2.5-Flash-T	31.7	23.8	37.3	29.5	21.5	11.5	23.0	20.8	13.9
<i>Close-source & Chat</i>									
Seed1.5-VL	54.0	46.2	59.3	51.8	57.0	38.5	42.4	48.0	20.0
GPT-4o	36.3	38.3	48.3	39.1	15.2	17.3	27.9	21.1	7.0
Claude-3.7-Sonnet	38.7	42.2	37.3	39.9	30.4	21.2	27.9	28.0	12.2
Kimi-Latest	31.3	29.6	38.1	31.8	20.9	3.8	20.0	18.1	0.9
<i>Open-source & Thinking</i>									
QVQ-72B-Preview	43.7	35.0	45.8	40.6	38.0	26.9	31.5	33.6	8.7
Virgo-72B	39.7	36.8	47.5	39.9	34.2	11.5	22.4	25.9	3.5
VL-Rethinker-72B	43.3	38.3	53.4	43.0	31.0	25.0	29.1	29.3	13.9
VL-Rethinker-7B	41.3	33.9	51.7	40.1	21.5	9.6	16.4	17.6	2.6
MM-Eureka-Qwen-32B	36.7	33.2	49.2	37.4	25.9	17.3	26.1	24.8	9.6
MM-Eureka-Qwen-7B	36.7	32.9	41.5	36.0	25.3	7.7	21.8	21.3	4.3
R1-VL-7B	31.7	24.9	34.7	29.5	13.3	5.8	12.7	12.0	0.9
Vision-R1-7B	32.3	29.6	33.9	31.5	18.4	9.6	16.4	16.3	4.3
R1-Onevision-7B-RL	34.3	27.8	31.4	31.2	15.2	9.6	12.1	13.1	0.9
Kimi-VL-A3B-T	33.0	27.4	34.7	31.1	34.2	13.5	14.5	22.7	6.1
OpenVLThinker-7B	38.7	28.2	41.5	35.0	15.8	7.7	11.5	12.8	0.9
LMM-R1-MGT-PerceReason	36.3	36.1	44.9	37.7	19.0	13.5	15.8	16.8	0.9
Mulberry	30.0	28.9	42.4	31.7	12.0	7.7	11.5	11.2	0.9
LlamaV-o1	26.7	25.3	29.7	26.6	14.6	1.9	7.9	9.9	0.9
<i>Open-source & Chat</i>									
Qwen2.5-VL-72B	41.0	34.3	55.1	40.7	34.8	23.1	26.1	29.3	9.6
Qwen2.5-VL-32B	44.0	30.0	50.0	39.4	34.2	15.4	27.3	28.5	12.2
Qwen2.5-VL-7B	39.0	30.0	41.5	35.8	17.1	15.4	18.2	17.3	3.5
Qwen2.5-VL-3B	33.3	30.7	46.6	34.5	19.6	11.5	13.9	16.0	0.0
Qwen2-VL-72B	35.3	36.5	46.6	37.7	16.5	11.5	18.8	16.8	1.7
Qwen2-VL-7B	32.7	28.9	45.8	33.4	12.0	9.6	12.7	12.0	0.9
Qwen2-VL-2B	30.0	30.3	31.4	30.4	8.9	1.9	6.1	6.7	0.0
InternVL3-78B	40.0	37.9	54.2	41.6	25.9	13.5	23.0	22.9	5.2
InternVL3-38B	36.7	32.1	48.3	36.8	27.8	13.5	17.0	21.1	2.6
InternVL3-8B	29.7	35.7	47.5	35.1	25.3	0.0	17.0	18.1	0.9
Molmo-72B	30.0	20.2	30.5	26.2	10.1	9.6	11.5	10.7	1.7
Molmo-7B-D	27.0	12.6	23.7	20.7	8.9	1.9	9.7	8.3	0.0
Molmo-7B-O	23.0	18.4	16.9	20.1	4.4	3.8	6.1	5.1	0.0
LLaVA-OV-72B	33.7	35.4	39.0	35.3	15.8	5.8	18.8	15.7	1.7
Kimi-VL-A3B	31.3	30.0	42.4	32.7	15.8	7.7	9.1	11.7	2.6

Table 13: Performance comparison of chat models with or w/o MCTS.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P& E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
Qwen2.5-VL-7B	22.2	18.2	21.9	35.1	36.1	31.4	27.5	20.9	26.8
+ MCTS	20.6	13.8	18.8	30.5	35.4	28.1	23.6	17.6	23.3

correct CoT outputs, but as model size increases, they begin to benefit noticeably from relatively accurate reasoning processes.

B.7 TOKEN USAGE OF THINKING MODELS

In Fig. 10, we present the average token length of different thinking models on MME-Reasoning. Overall, there is a clear trend indicating that better model performance is often associated with longer reasoning paths. However, we also observe diminishing returns between output length and performance in both open-source and closed-source models.

Additionally, although current rule-based reinforcement learning (RL) models show a promising trend of increased output length during training, no significant length gains were observed on MME-

Table 14: Performance comparison of SoTA chat models with or w/o CoT prompt.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P&E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
Qwen2.5-VL-7B	22.2	18.2	21.9	35.1	36.1	31.4	27.5	20.9	26.8
+ CoT prompt	20.3	18.5	17.9	33.3	38.9	28.3	21.4	23.6	24.7
Qwen2.5-VL-32B	32.2	26.8	24.4	39.0	52.1	40.5	27.5	29.6	33.2
+ CoT prompt	29.0	24.6	23.3	40.8	52.1	40.1	28.7	26.4	32.3
Qwen2.5-VL-72B	31.7	25.1	27.2	37.9	53.5	39.0	32.3	29.9	34.1
+ CoT prompt	32.5	26.2	25.1	37.2	52.8	37.5	30.8	30.0	33.0
InternVL3-8B	19.5	19.6	22.6	31.6	41.0	28.1	29.9	21.4	26.4
+ CoT prompt	21.1	16.3	20.2	31.6	38.2	31.2	26.9	16.8	25.2
InternVL3-38B	23.0	18.5	23.0	38.3	41.7	33.5	29.0	22.1	28.4
+ CoT prompt	28.7	24.3	28.6	38.3	48.6	37.5	32.9	26.9	32.7
InternVL3-78B	26.0	24.0	26.5	41.8	50.0	35.1	33.8	27.1	32.1
+ CoT prompt	29.0	22.9	27.0	40.8	48.6	36.6	35.1	26.9	32.9

Model	CoT Prompt
Qwen2.5-VL	Let’s think step by step.
InternVL3	Answer the preceding question. The last line of your response should follow this format: 'Answer: \$FINAL_ANSWER' (without quotes), where 'FINAL_ANSWER' is your conclusion based on the reasoning provided. If you are uncertain or the problem is too complex, make a reasoned guess based on the information provided. Avoid repeating steps indefinitely—provide your best guess even if unsure. Think step by step logically, considering all relevant information before answering.

Table 15: Chain-of-Thought Prompts for Different Models

Table 16: Performance of Caption + SoTA Reasoning LLMs. We use GPT-4o to generate caption of each image in MME-Reasoning.

Model	Model Capability					Reasoning Type			AVG.
	CAL.	P&E.	PA.	S&T.	CCA.	DED.	IND.	ABD.	
QwQ-32B	48.5	32.9	39.1	37.6	53.5	44.4	45.6	35.9	41.9
DeepSeek-R1	56.9	40.0	41.6	41.8	58.3	53.8	43.8	41.5	46.9

Reasoning. This limitation may stem from the limited types and inappropriate complexity of the reasoning tasks. Therefore, exploring how different types of reasoning tasks can better stimulate the effectiveness of RL in reasoning may be a valuable direction for future research.

B.8 RESULTS OF CAPTIONER & LLMs

We used GPT-4o as the captioner to generate visual descriptions for each question as a substitute for the images. Then we evaluated existing LLMs with "thinking mode," and the results are presented in Tab. 16. As shown in the results, even when only indirectly perceiving image content through textual descriptions, QwQ (Qwen Team, 2025b) and R1 (DeepSeek-AI, 2025) achieved impressive scores of 41.9 and 46.9 respectively—surpassing even Claude-3.7-Sonnet-Thinking. These findings indicate that there is still substantial room for improvement in extending long-term reasoning capabilities from LLMs to the multimodal domain. This gap may be due, in part, to degradation in the foundational model’s capabilities during the vision-language alignment process. Additionally, the diversity of reasoning tasks specific to multimodal settings has yet to be thoroughly explored.

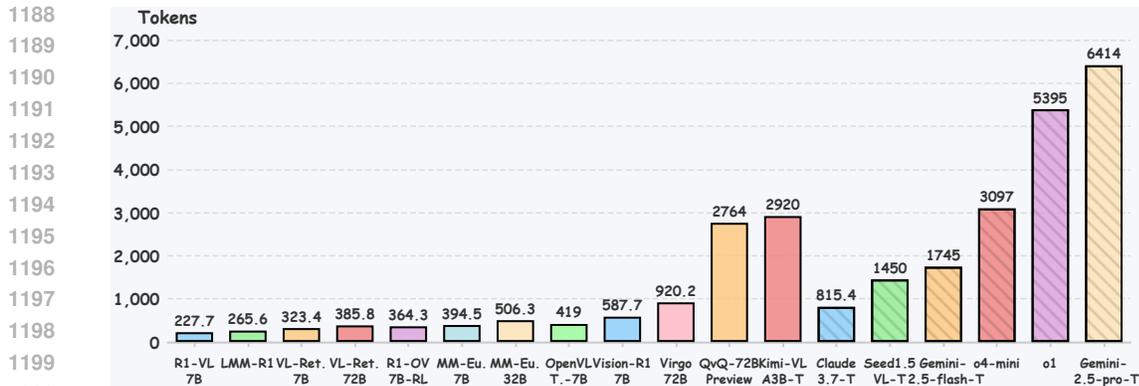


Figure 10: Average token usage of open & closed-source thinking models on MME-Reasoning.

Table 17: Response token length on different datasets.

Model	MathVista	MathVerse	MME-Reasoning
Qwen2.5-VL-7B	209.5	207.6	442.8
GPT-4o	162.6	157.3	328.0
Claude-3.7-Sonnet-T	519.4	563.2	979.2

B.9 RESPONSE LENGTH OF DIFFERENT BENCHMARKS

We report the average response lengths of three representative models across different datasets in Tab. 17. Results show that responses on MME-Reasoning are significantly longer than those on previous reasoning benchmarks, indicating its challenging and rigorous demands on model reasoning.

C DETAILS OF ANNOTATION

C.1 OVERALL ANNOTATION PROCESS

Our overall annotation process is as follows:

- Each problem is independently evaluated by at least two annotators.
- If the first two annotators assign different annotations, the problem was sent to a third annotator for a final decision.
- The final annotation is only assigned if at least two out of the three annotators are in agreement.

C.2 DIFFICULT ANNOTATION

For each question, we assign a difficulty label: *Easy*, *Medium*, or *Hard*, based on the cognitive load required to solve it. The labeling criteria are as follows:

- *Easy*: The question typically has a straightforward and quick solution that can be correctly answered by a human expert within 2 minutes.
- *Medium*: The question generally requires some reasoning steps and one to two rounds of trial and reflection, and can be correctly answered by a human expert within 2 to 5 minutes.
- *Hard*: The question usually requires more than two attempts and reflections, or involves the use of tools such as auxiliary lines or drafts to support the thought process. It may or may not be solved by a human expert within 10 minutes.

1242 C.3 REASONING TYPE ANNOTATION

1243
1244 For each question, we assign a reasoning type label: *Deductive*, *Inductive*, or *Abductive*, based on
1245 the dominant reasoning method required in its solution. The labeling criteria are as follows:

- 1246 • ***Deductive***: Involves deriving a necessary conclusion from given premises and general rules
1247 through step-by-step inference. Examples include math problems, physics problems, and
1248 certain puzzles.
- 1249 • ***Inductive***: Involves observing specific phenomena, summarizing general patterns or rules,
1250 and extrapolating based on those patterns. Examples include figure series and analogy
1251 questions.
- 1252 • ***Abductive***: Involves forming hypotheses or explanations based on known phenomena and
1253 then verifying them. These problems typically have a large solution space. Examples
1254 include Sudoku, mate-in-one chess problems, circuit fault analysis, biological pedigree
1255 analysis, and some puzzles.

1256
1257 It should be noted that although the solutions to some puzzles, such as Sudoku, can theoretically be
1258 derived through deductive reasoning, in the actual process of human reasoning, we often resort to
1259 assuming a certain move and then verifying its validity. This hypothesis–verification–backtracking
1260 mechanism leads us to consider these a form of abductive reasoning.

1261 C.4 CAPABILITY ANNOTATION

1262
1263 Based on the both empirical analysis of the results, especially failure cases, and cognitive the-
1264 ory Gardner (2011), we identify five capabilities related to MLLMs’ reasoning capability (*i.e.*, cal-
1265 culation, causal chain analysis, pattern analysis, planning and exploration, spatial and temporal).
1266 For each question, we also assign one or more capability labels based on the primary abilities being
1267 tested. The available labels are: *Pattern Analysis*, *Planning and Exploring*, *Spatial and Temporal*,
1268 *Calculation*, and *Causal Chain Analysis*. A question may have multiple capability labels. The
1269 labeling criteria are as follows:

- 1270 • ***Pattern analysis***: Requires identifying patterns in shape, color, size, or other visual features
1271 within the image.
- 1272 • ***Planning and exploring***: Requires explicit planning of the answering process, involving
1273 exploration within solution space and iterative verification or reflection.
- 1274 • ***Spatial and Temporal***: Requires understanding spatial relationships or temporal sequences
1275 represented in the visual input.
- 1276 • ***Calculation***: Involves performing numerical calculations based on given quantitative con-
1277 ditions to arrive at a correct result.
- 1278 • ***Causal Chain Analysis***: Requires reasoning about causal relationships across multiple
1279 nodes based on limited information, or understanding dynamic processes in the problem
1280 and identifying key events.

1282 D DETAILS OF IMPLEMENTATION

1283 D.1 IMPLEMENTATION DETAILS

1284
1285 Some of the data in MME-Reasoning are sourced from ScanNet (Dai et al., 2017), Ark-
1286 itscenescenes (Baruch et al., 2021), VideoMME (Fu et al., 2024), MM-IQ (Cai et al., 2025), Puz-
1287 zleVQA (Chia et al., 2024). We further filter most of the data and reformulate the questions. We
1288 use gpt-4o-mini to extract the answer of all responses and judge the answer of free-form questions.
1289 The cost fluctuates with the length of the MLLM’s response. As an example, extracting and judg-
1290 ing the response of Qwen2.5-VL-72B costs around \$0.1. We use VLMEvalKit¹ to evaluate all the
1291 models. For models larger than 30B, we use vllm² to reduce the inference time. All experiments are
1292 conducted on A100 GPUs except experiments on closed-source models.

1293
1294
1295 ¹<https://github.com/open-compass/VLMEvalKit>

²<https://github.com/vllm-project/vllm>

1296 E DETAILS OF EVALUATION

1297

1298 E.1 PROMPTS FOR ANSWER EXTRACTION

1299

1300 We list our answer extraction prompts from Fig. 11 to Fig. 21 including:

1301

- 1302 • Fig. 11: Prompt for tasks answering in ‘id : answer’ format.
- 1303 • Fig. 12: Prompt for tasks answering in ‘coordinates’ format.
- 1304 • Fig. 13: Prompt for tasks answering in ‘formula’ format.
- 1305 • Fig. 14: Prompt for multiple-choice tasks.
- 1306 • Fig. 15: Prompt for points24 tasks.
- 1307 • Fig. 16: Prompt for hashi puzzles.
- 1308 • Fig. 17: Prompt for sudoku_4x4 puzzles.
- 1309 • Fig. 18: Prompt for sudoku_6x6 puzzles.
- 1310 • Fig. 19: Prompt for skyscraper puzzles.
- 1311 • Fig. 20: Prompt for yinyang puzzles.
- 1312 • Fig. 21: Prompt for free-form tasks.

1313

1314

1315

1316

1317

1318

1319

1320

1321

1322

1323

1324

1325

1326

1327

1328

1329

1330

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

F EXAMPLES OF MME-REASONING

We further provide additional case studies as shown from Fig. 22 to Fig. 52, showing both correct and incorrect responses by MLLMs (*e.g.*, select from GPT-4o, Qwen2.5-VL-72B, o4-mini, Seed1.5-VL-Thinking, and Gemini-2.5-Pro-Thinking). In each figure, we show the original questions, reasoning types, difficulty levels, and model responses. Overall, we find that “thinking models” demonstrate stronger abilities in exploration, judgment, and reflection. However, it still struggles to arrive at correct answers for many reasoning problems that are simple for humans, indicating that the model’s reasoning ability still needs further improvement. Moreover, the number of tokens consumed by the reasoning model increases rapidly. Therefore, future research should also focus on balancing both the reasoning ability and efficiency of the model.

1327

1328

1329

1330

1331

1332

1333

1334

1335

1336

1337

1338

1339

1340

1341

1342

1343

1344

1345

1346

1347

1348

1349

G LIMITATION

Despite our best efforts to cover a wide range of multimodal reasoning question types, it remains challenging to comprehensively collect all possible types of reasoning problems that occur in real-world scenarios. This is primarily because gathering and curating high-quality reasoning questions is often a time-consuming and labor-intensive process. Future work is needed to further enrich the diversity of question types and optimize dataset coverage.

1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
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

Please read the following example. Then extract the answer from the model response and type it at the end of the prompt.
Example:
Question: Each cycle represents a number. You need to find out what the three numbers are. Give a possible answer in the format 'cycle id:number'
Model Response: The possible answer is: A:5, B:1, C:2
Extracted answer (json format):

```
{
  "A":5,
  "B":1,
  "C":2
}
```

Please extract the answer for the following response:
Question: {question}
Model Response: {response}
You should only output the json without any other texts.

Figure 11: Prompt for tasks answering in 'id : answer' format.

Please read the following example. Then extract the answer from the model response and type it at the end of the prompt.
Example1:
Question: According to the clues, find the corresponding position. Answer in '(row id (A-C), column id (1-3))' format.
Model Response: The possible answer is: (A, 1)
Extracted answer (json format):

```
[
  {
    "row": "A",
    "column": 1
  }
]
```

Example2:
Question: According to the clues, find the two corresponding position. Answer in '(row id (A-C), column id (1-3))' format.
Model Response: The possible answer is: (A, 1), (B, 3)
Extracted answer (json format):

```
[
  {
    "row": "A",
    "column": 1
  },
  {
    "row": "B",
    "column": 3
  }
]
```

Please extract the answer for the following response:
Question: {question}
Model Response: {response}
You should only output the json without any other texts.

Figure 12: Prompt for tasks answering in 'coordinates' format.

1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415

Please extract the mathematical formula from the following model response and type it at the end of the prompt.
Example:
Question: What is the right equation to solve the problem?
Model Response: The right equation to solve the problem is: $2 + 3 = 7$
Extracted answer (json format):
{
 "equation": " $2 + 3 = 7$ "
}

Please extract the answer for the following response:
Question: {question}
Model Response: {response}
You should only output the json without any other texts.

Figure 13: Prompt for tasks answering in 'formula' format.

1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434

Please read the following example. Then extract the answer from the model response and type it at the end of the prompt.
Example1:
Question: Which answer is right?\n A.1\n B.2\n C.3\n D.4\n Please answer the question and provide the correct option letter, e.g., A, B, C, D, at the end.
Model Response: The possible answer is: A
Extracted answer: A
Example2:
Question: Which answer is right?\n A.1\n B.2\n C.3\n D.4\n Please answer the question and provide all correct option letter, e.g., A, B, C, D, at the end. Find all possible answers.
Model Response: The possible answer is: A, C
Extracted answer: [A, C]

Please extract the answer for the following response:
Question: {question}
Model Response: {response}
Extracted answer:
You should only output the answer without any other texts.

Figure 14: Prompt for multiple-choice tasks.

1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454

Please read the following examples. Then extract the final formula from the answer to the 24-point game, and type it at the end of the prompt. In the cards, K stands for 13, Q stands for 12, J stands for 11, and A stands for 1. Note you need to use * to represent multiplication sign, / to represent division sign.
Example1:
Question: Given four playing cards (A, 8, 9, K), each with a value, use any combination of addition, subtraction, multiplication, and division to make the number 24. You must use each card exactly once. Give the final answer as a formula.
Model Response: The possible answer is $(K - 9 - A) \times 8 = 24$
Extracted answer: $(13-9-1)*8=24$
Example2:
Question: Given four playing cards (3, 8, 9, A), each with a value, use any combination of addition, subtraction, multiplication, and division to make the number 24. You must use each card exactly once. Give the final answer as a formula.
Model Response: The possible answer is $9 \div 3 \times A \times 8 = 24$
Extracted answer: $9/3*1*8=24$

Please extract the final formula from for the following response:
Question: {question}
Model Response: {response}
Extracted answer:
You should only output the final formula from without any other texts.

Figure 15: Prompt for points24 tasks.

1455
1456
1457

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

Extract all bridge connections from the Hashi puzzle solution text and format them as a structured JSON list. Follow these rules:

- Input**:
 - `solution`: Text describing bridges between islands using various formats (e.g., "c1 - c3", "a1到g1", "between b2 and b4").
- Output Requirements**:
 - Return a JSON list of dictionaries in this format:


```
```JSON
 [{"start": "a1", "end": "b1", "number": 2}], ...]
          ```
```
 - Include ALL bridges explicitly described in `solution`.
 - Use 1-based row numbers and letter-based columns (e.g., "c3" not "3c").
 - Normalize coordinate formats (e.g., "(1,c)" → "c1", "d,4" → "d4").
- Parsing Rules**:
 - Capture bridge counts (1 or 2) from phrases like:
 - "two bridges"
 - "1 bridge"
 - "double bridge"
 - Handle directional phrases:
 - "from X to Y"
 - "X connects to Y"
 - "X-Y bridge"
- Edge Cases**:
 - The bridge corresponding to the following plot will be skipped directly:
 - No bridges can be parsed
 - Ambiguous connections (unclear start/end)
 - Invalid coordinate formats

Response Format:

- Return ONLY the JSON list.
- Do not include any additional text, explanations, or formatting beyond the JSON list.

Examples:

Solution:
"Draw two bridges from a1 to b1, and one bridge between f6-f8"
Output:
[[{"start": "a1", "end": "b1", "number": 2}, {"start": "f6", "end": "f8", "number": 1}]]

Solution:
"Connect c3 with d3 using a double bridge"
Output:
[[{"start": "c3", "end": "d3", "number": 2}]]

Solution:
{response}
Output:

Figure 16: Prompt for hashi puzzles.

1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565

Extract the final answer from the given solution. The solution will contain a detailed solution to a Sudoku-like puzzle, including step-by-step explanations and a final filled grid representation. Your task is to identify and extract only the final answer, which is presented as a matrix (list of lists or equivalent) at the end of the solution.

Requirements:

1. Only return the final filled matrix, formatted as a JSON list of lists.
2. Do not include any additional text, explanations, or formatting beyond the matrix itself.
3. The final answer usually follows phrases such as 'Final Filled Grid' or 'Summary as a Single Filled Matrix'.
4. If the content in the given solution cannot be extracted as a matrix, return 'False' directly

Example 1:

Solution:

To solve the given Jigsaw Sudoku puzzle, we need to ensure that each row, each column, and each 2x2 sub-grid contains the digits 1 through 4 exactly once. Let's fill in the grid step by step.

Step-by-Step Solution:

1. Identify the given numbers and their positions:
 - Top row: 1, 3
 - Second row: 3
 - Third row: (empty)
 - Fourth row: 2, 3, 1

2. Fill in the top row:
 - The top row already has 1 and 3. The remaining digits are 2 and 4.
 - The first column already has a 3, so the first cell in the top row cannot be 3. Therefore, it must be 2.
 - The fourth column already has a 3, so the fourth cell in the top row cannot be 3. Therefore, it must be 4.

Final Filled Matrix:

```
[[2, 1, 4, 3], [1, 4, 2, 3], [4, 2, 1, 3], [2, 3, 4, 1]]
```

Output:

```
[[2, 1, 4, 3], [1, 4, 2, 3], [4, 2, 1, 3], [2, 3, 4, 1]]
```

Example 2:

Solution:

1. Identify the given numbers and their positions:
 - Top row: 1, 3
 - Second row: 3
 - Third row: (empty)
 - Fourth row: 2, 3, 1

2. Fill in the top row:
 - The top row already has 1 and 3. The remaining digits are 2 and 4.
 - The first column already has a 3, so the first cell in the top row cannot be 3. Therefore, it must be 2.
 - The fourth column already has a 3, so the fourth cell in the top row cannot be 3. Therefore, it must be 4.

3. Fill in the second row:
 - The second row already has a 3. The remaining digits are 1, 2, and 4.
 - The first column already has a 2 and a 3, so the first cell in the second row cannot be 2 or 3. Therefore, it must be 1.
 - The second column already has a 1, so the second cell in the second row cannot be 1. Therefore, it must be 4.
 - The fourth column already has a 3 and a 4, so the fourth cell in the second row cannot be 3 or 4. Therefore, it must be 2.

4. Fill in the third row:
 - The third row is empty. The remaining digits are 1, 2, 3, and 4.
 - The first column already has a 2 and a 1, so the first cell in the third row cannot be 2 or 1. Therefore, it must be 4.
 - The second column already has a 1 and a 4, so the second cell in the third row cannot be 1 or 4. Therefore, it must be 2.
 - The third column already has a 4 and a 2, so the third cell in the third row cannot be 4 or 2. Therefore, it must be 1.
 - The fourth column already has a 3 and a 2, so the fourth cell in the third row cannot be 3 or 2. Therefore, it must be 3.

5. Fill in the fourth row:
 - The fourth row already has 2, 3, and 1. The remaining digit is 4.

Final Filled Matrix:

```
python [2, 1, 4, 3], [1, 4, 2, 3], [4, 2, 1, 3], [2, 3, 4, 1]
```

Summary:

```
The completed Jigsaw Sudoku puzzle is: python [2, 1, 4, 3], [1, 4, 2, 3], [4, 2, 1, 3], [2, 3, 4, 1]
```

Output:

```
[[2, 1, 4, 3], [1, 4, 2, 3], [4, 2, 1, 3], [2, 3, 4, 1]]
```

Solution:

```
{response}
```

Output:

Figure 17: Prompt for sudoku_4x4 puzzles.

1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619

Extract the final answer from the given solution. The solution will contain a detailed solution to a Sudoku-like puzzle, including step-by-step explanations and a final filled grid representation. Your task is to identify and extract only the final answer, which is presented as a matrix (list of lists or equivalent) at the end of the solution.

Requirements:

1. Only return the final filled matrix, formatted as a JSON list of lists.
2. Do not include any additional text, explanations, or formatting beyond the matrix itself.
3. The final answer usually follows phrases such as 'Final Filled Grid' or 'Summary as a Single Filled Matrix'.
4. If the content in the given solution cannot be extracted as a matrix, return 'False' directly

Example 1:

Solution:

To solve the given Jigsaw Sudoku puzzle, we need to ensure that each row, each column, and each 2x3 sub-grid contains the digits 1 through 6 exactly once. Let's solve it step by step.

Step-by-Step Solution:

1. Identify the sub-grids and initial constraints: The grid is divided into 2x3 sub-grids. We need to fill in the empty cells while satisfying the Sudoku rules.

2. Fill in the first row: The first row has the digits 2 and 4 already placed. The remaining digits are 1, 3, 5, and 6. Considering the constraints of the sub-grids and columns, we can place the digits as follows: First row: [5, 3, 2, 1, 6, 4].

3. Fill in the second row: The second row has the digits 6 and 1 already placed. The remaining digits are 2, 3, 4, and 5. Considering the constraints, we can place the digits as follows: Second row: [4, 6, 1, 5, 3, 2].

4. Fill in the third row: The third row has the digits 1, 3, and 6 already placed. The remaining digits are 2, 4, and 5. Considering the constraints, we can place the digits as follows: Third row: [1, 3, 5, 6, 2, 4].

5. Fill in the fourth row: The fourth row has the digits 6, 3, and 1 already placed. The remaining digits are 2, 4, and 5. Considering the constraints, we can place the digits as follows: Fourth row: [2, 5, 6, 4, 3, 1].

6. Fill in the fifth row: The fifth row has the digits 4 and 2 already placed. The remaining digits are 1, 3, 5, and 6. Considering the constraints, we can place the digits as follows: Fifth row: [3, 1, 4, 2, 5, 6].

7. Fill in the sixth row: The sixth row has the digits 6 and 3 already placed. The remaining digits are 1, 2, 4, and 5. Considering the constraints, we can place the digits as follows: Sixth row: [6, 2, 1, 3, 4, 5].

Final Filled Matrix:

```
python
[5, 3, 2, 1, 6, 4],
[4, 6, 1, 5, 3, 2],
[1, 3, 5, 6, 2, 4],
[2, 5, 6, 4, 3, 1],
[3, 1, 4, 2, 5, 6],
[6, 2, 1, 3, 4, 5]
```

Output:

```
[[5, 3, 2, 1, 6, 4],[4, 6, 1, 5, 3, 2],[1, 3, 5, 6, 2, 4],[2, 5, 6, 4, 3, 1],[3, 1, 4, 2, 5, 6],[6, 2, 1, 3, 4, 5]]
```

Example 2:

Solution:

To solve this 6x6 Jigsaw Sudoku puzzle, we need to ensure that:

1. Each row contains the digits 1-6 exactly once.
2. Each column contains the digits 1-6 exactly once.
3. Each 2x3 sub-grid contains the digits 1-6 exactly once.

Here is the completed solution:

Completed Sudoku Grid:

```
python
[1, 3, 4, 2, 5, 6],
[6, 2, 5, 1, 3, 4],
[5, 6, 2, 3, 4, 1],
[3, 4, 6, 5, 1, 2],
[4, 5, 1, 6, 2, 3],
[2, 1, 3, 4, 6, 5]
```

Output:

```
[[1, 3, 4, 2, 5, 6],[6, 2, 5, 1, 3, 4],[5, 6, 2, 3, 4, 1],[3, 4, 6, 5, 1, 2],[4, 5, 1, 6, 2, 3],[2, 1, 3, 4, 6, 5]]
```

Solution:

{response}

Output:

Figure 18: Prompt for sudoku_6x6 puzzles.

1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673

Extract the final answer from the given solution. The solution will contain a detailed solution to a Sudoku-like puzzle, including step-by-step explanations and a final filled grid representation. Your task is to identify and extract only the final answer, which is presented as a matrix (list of lists or equivalent) at the end of the solution.

Requirements:

1. Only return the final filled matrix, formatted as a JSON list of lists.
2. Do not include any additional text, explanations, or formatting beyond the matrix itself.
3. The final answer usually follows phrases such as 'Final Filled Grid' or 'Summary as a Single Filled Matrix'.
4. If the content in the given solution cannot be extracted as a matrix, return 'False' directly

Example 1:

Solution:

To solve the 3x3 Skyscrapers puzzle, we need to place the numbers 1, 2, and 3 in each row and column while adhering to the given clues. Let's break down the solution step by step. ### Step-by-Step Solution: \n\n1. **Analyze the clues:** \n - Top row: 3, 1, 2 \n - Left column: 2, 2, 1 \n - Right column: 2, 1, 3 \n - Bottom row: 1, 2, 2 \n\n2. **Start with the top row:** \n - The top row has a clue of 3 on the left, meaning all three skyscrapers are visible. This implies the order must be 1, 2, 3 from left to right. \n - The top row is: [1, 2, 3] \n\n3. **Analyze the left column:** \n - The left column has a clue of 2 at the top, meaning two skyscrapers are visible. Given the top row is [1, 2, 3], the second row must start with a 3 to block the view of the third row. \n - The left column is: [1, 3, 2] \n\n4. **Analyze the right column:** \n - The right column has a clue of 3 at the bottom, meaning all three skyscrapers are visible. This implies the order must be 1, 2, 3 from top to bottom. \n - The right column is: [3, 2, 1] \n\n5. **Fill in the remaining cells:** \n - The second row has a clue of 2 on the left, and we already have a 3 in the first column. The remaining cells must be 1 and 2 in some order. Given the right column clue of 2, the second row must be [3, 1, 2]. \n - The third row has a clue of 1 on the left, meaning the first cell must be 2 to block the view of the other cells. The remaining cells must be 3 and 1 in some order. Given the right column clue of 3, the third row must be [2, 3, 1]. \n\n### Final Solution: \n\nThe completed 3x3 Skyscrapers puzzle is: \n\n```\n1 2 3\n3 1 2\n2 3 1\n```\n\n### Summary as a single filled matrix: \n\n```\npython\n[[1, 2, 3], [3, 1, 2], [2, 3, 1]]\n```\n

Output:

[[1, 2, 3], [3, 1, 2], [2, 3, 1]]

Example 2:

Solution:

To solve the 4x4 Skyscrapers puzzle, we need to place the numbers 1, 2, 3, and 4 in each row and column while adhering to the given clues. Let's break down the solution step by step. ### Step-by-Step Solution: \n\n1. **Analyze the clues:** \n - Top row: Clue '2' on the left means we can see 2 skyscrapers from the left. \n - Bottom row: Clue '4' on the right means we can see 4 skyscrapers from the right. \n - Left column: Clue '2' on the top means we can see 2 skyscrapers from the top. \n - Right column: Clue '2' on the bottom means we can see 2 skyscrapers from the bottom. \n\n2. **Start with the bottom row:** \n - Since we can see 4 skyscrapers from the right, the bottom row must be in ascending order from left to right: [1, 2, 3, 4]. \n\n3. **Fill the bottom row:** \n - Row 4: [1, 2, 3, 4] \n\n4. **Analyze the top row:** \n - We can see 2 skyscrapers from the left, so the tallest skyscraper (4) must be in the second position from the left to block the view of the third and fourth skyscrapers. \n - Possible configurations: [3, 4, 1, 2] or [2, 4, 1, 3] \n\n5. **Analyze the left column:** \n - We can see 2 skyscrapers from the top, so the tallest skyscraper (4) must be in the second position from the top to block the view of the third and fourth skyscrapers. \n - Possible configurations: [3, 4, 1, 2] or [2, 4, 1, 3] \n\n6. **Combine the clues:** \n - Top row: [3, 4, 1, 2] \n - Left column: [3, 4, 1, 2] \n\n7. **Fill the remaining cells:** \n - Row 1: [3, 4, 1, 2] \n - Row 2: [4, 1, 2, 3] \n - Row 3: [2, 3, 4, 1] \n - Row 4: [1, 2, 3, 4] \n\n### Final Solution: \n\n```\n[[3, 4, 1, 2], [4, 1, 2, 3], [2, 3, 4, 1], [1, 2, 3, 4]]\n```\n\nThis matrix satisfies all the given clues and constraints of the Skyscrapers puzzle.

Output:

[[3, 4, 1, 2], [4, 1, 2, 3], [2, 3, 4, 1], [1, 2, 3, 4]]

Solution:

{response}

Output:

Figure 19: Prompt for skyscraper puzzles.

1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727

Extract the final answer from the given solution. The solution will contain a detailed solution to a Sudoku-like puzzle, including step-by-step explanations and a final filled grid representation. Your task is to identify and extract only the final answer, which is presented as a matrix (list of lists or equivalent) at the end of the solution.

Requirements:

1. Only return the final filled matrix, formatted as a JSON list of lists.
2. Do not include any additional text, explanations, or formatting beyond the matrix itself.
3. The final answer usually follows phrases such as 'Final Filled Grid' or 'Summary as a Single Filled Matrix'.
4. If the content in the given solution cannot be extracted as a matrix, return 'False' directly

Example 1:

Solution:

To solve the Yin-Yang puzzle, we need to place black and white stones at the intersections of the grid lines while adhering to the given rules. Let's analyze the current state of the grid and determine the placement of the remaining stones.

Step-by-Step Solution:

1. Initial Setup: The grid is a 6x6 matrix. Some cells already have black (1) and white (0) stones.
2. Rule Analysis: All black stones must be orthogonally connected. All white stones must be orthogonally connected. No 2x2 region can be monochromatic. Existing stones cannot be moved.
3. Placement Strategy: Start by ensuring the connectivity of black and white stones. Avoid creating monochromatic 2x2 regions.
4. Detailed Placement:
 - Top-left corner (a1): Place a white stone to avoid a monochromatic 2x2 region with the black stone at (b1).
 - Top-right corner (f1): Place a white stone to avoid a monochromatic 2x2 region with the black stone at (e1).
 - Middle section: Ensure connectivity of black and white stones while avoiding monochromatic 2x2 regions.

Final Matrix:

After placing the stones according to the rules, the final matrix is:

```
python
[
  [0, 1, 0, 1, 0, 0],
  [0, 0, 1, 1, 0, 0],
  [1, 0, 1, 0, 1, 0],
  [0, 1, 0, 1, 0, 1],
  [0, 0, 1, 0, 1, 0],
  [1, 0, 0, 1, 0, 0]
]
```

Summary:

The completed solution for the Yin-Yang puzzle is represented as a single filled matrix:

```
python
[
  [0, 1, 0, 1, 0, 0],
  [0, 0, 1, 1, 0, 0],
  [1, 0, 1, 0, 1, 0],
  [0, 1, 0, 1, 0, 1],
  [0, 0, 1, 0, 1, 0],
  [1, 0, 0, 1, 0, 0]
]
```

Output:

```
[[0, 1, 0, 1, 0, 0],[0, 0, 1, 1, 0, 0],[1, 0, 1, 0, 1, 0],[0, 1, 0, 1, 0, 1],[0, 0, 1, 0, 1, 0],[1, 0, 0, 1, 0, 0]]
```

Example 2:

Solution:

To solve the Yin-Yang puzzle, we need to ensure that the rules are followed:

1. All black stones form a single orthogonally connected group.
2. All white stones form a single orthogonally connected group.
3. No 2x2 region is monochromatic.
4. The positions of the pre-placed stones cannot be changed.

Step-by-Step Solution:

1. Initial Setup: The given grid is a 6x6 grid with some pre-placed black and white stones. We'll represent the grid as follows: "1" for black stones, "0" for white stones. Empty cells will be filled as we solve the puzzle.
2. Pre-placed Stones: From the diagram: Black stones ("1") are at: (a1, b1, c1, c2, d2, d3, e3). White stones ("0") are at: (a3, a4, a5, b4, b5, c4, d4, e4).
3. Solving the Puzzle: We will fill the remaining cells while ensuring the rules are satisfied.
4. Final Solution: After solving, the completed grid is as follows:

```
python
[
  [1, 1, 0, 0, 0, 0],
  [1, 0, 1, 0, 0, 0],
  [0, 1, 0, 1, 0, 1],
  [0, 0, 0, 1, 0, 1],
  [0, 0, 0, 0, 0, 1],
  [1, 0, 0, 0, 1, 1]
]
```

Summary:

The solution as a 6x6 matrix is:

```
python
[
  [1, 1, 0, 0, 0, 0],
  [1, 0, 1, 0, 0, 0],
  [0, 1, 0, 1, 0, 1],
  [0, 0, 0, 1, 0, 1],
  [0, 0, 0, 0, 0, 1],
  [1, 0, 0, 0, 1, 1]
]
```

Output:

```
[[1, 1, 0, 0, 0, 0],[1, 0, 1, 0, 0, 0],[0, 1, 0, 1, 0, 1],[0, 0, 0, 1, 0, 1],[0, 0, 0, 0, 0, 1],[1, 0, 0, 0, 1, 1]]
```

Solution:

{response}

Output:

Figure 20: Prompt for yinyang puzzles.

1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781

Please read the following example. Then extract the answer from the model response and type it at the end of the prompt.

Example1:
Question: What value should be? Please answer the question and provide the final value, e.g., 2, 2.10, -3, at the end.
Model Response: The possible value is -4
Extracted answer: -4

Example2:
Question: What value should be? Please answer the question and provide the final value, e.g., 2, 2.10, -3, at the end.
Model Response: The possible value is $2\sqrt{3}$
Extracted answer: $2\sqrt{3}$

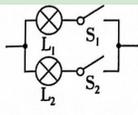
Example3:
Question: What value should be? Please answer the question and provide the final value, e.g., 2, 2.10, -3, at the end.
Model Response: The possible value is 3.28
Extracted answer: 3.28

Please extract the answer for the following response:
Question: {question}
Model Response: {response}
Extracted answer:
You should only output the answer without any other texts.

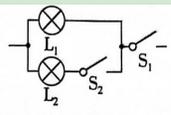
Figure 21: Prompt for free-form tasks.

1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835

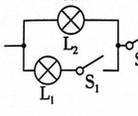




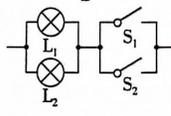
A



B



C



D

Question: A sealed box with two switches S_1 , S_2 and two lamps L_1 , L_2 visible on its surface. To investigate how the switches and lamps are connected inside the sealed box, performing the following operations: First, close S_1 , only lamp L_1 lights up; then close S_2 , both lamps light up; finally, open S_1 , both lamps go out. The connection method that matches the above experimental results is ().

Reasoning type: Abductive
Difficulty level: Medium

A. Diagram A B. Diagram B
C. Diagram C D. Diagram D



o4-mini

The only network that reproduces
 S_1 closed, S_2 open \rightarrow only L_1 on S_1 & S_2 closed \rightarrow both on
 S_1 open, S_2 closed \rightarrow both off
 is the "ladder" form in which the two lamps sit in series, with S_1 shunting (i.e. in parallel with) L_2 and S_2 shunting L_1 . **Among the four drawings that is diagram D.**



Gemini-2.5-Pro-Thinking

Option A: L_1 is in series with S_1 . L_2 is in series with S_2 . These two branches are in parallel.

... * Step 3: Open S_1 (S_2 remains closed). The L_1 - S_1 branch is broken, so L_1 goes out. However, the L_2 - S_2 branch is still complete, so current flows through L_2 , and L_2 remains lit. This contradicts the third observation. Therefore, Option A is incorrect.

Option B: L_1 and L_2 are connected in parallel. S_2 is in series with L_2 . S_1 is in series with the entire parallel combination of L_1 and the (L_2 - S_2) branch.

* Step 1: Close S_1 (S_2 is open). Current flows through S_1 , then reaches the junction. It can flow through L_1 . It cannot flow through the L_2 - S_2 branch because S_2 is open. So, only L_1 lights up. This matches the first observation....

* Step 3: Open S_1 . Opening S_1 breaks the main circuit path before the current can reach either L_1 or the L_2 - S_2 branch. No current flows through either lamp. Both L_1 and L_2 go out. This matches the third observation. Therefore, Option B is correct.

Option C: L_1 is in series with S_1 . This branch is in parallel with L_2 . S_2 is in series with this entire parallel combination.

* Step 1: Close S_1 (S_2 is open). Since S_2 is in the main circuit path and is open, no current can flow through the circuit. Neither L_1 nor L_2 lights up. This contradicts the first observation. * Therefore, Option C is incorrect.

Option D: L_1 and L_2 are in parallel. S_1 and S_2 are in parallel. The parallel lamp combination is in series with the parallel switch combination.

* Step 1: Close S_1 (S_2 is open). Current flows through the S_1 path, then splits to flow through both L_1 and L_2 . Both L_1 and L_2 light up. This contradicts the first observation.

* Therefore, Option D is incorrect.

Final Answer: The final answer is \boxed{B}

Figure 22: An example of circuit analysis in which the model needs to infer the circuit connection based on the observed phenomena.

1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889



Question: When you are next to the TV in room A facing the door, navigate to the sofa by completing the route: 1. Go straight out of the bedroom door, 2. [?], 3. Go straight to the dining table, 4. [?], 5. Go straight to the sofa.

A. Turn left, Turn right
B. Turn right, Turn left
C. Turn left, Turn left
D. Turn right, Turn right

Reasoning type: Deductive
Difficulty level: Hard



GPT-4o

To navigate from the TV in room A to the sofa:

1. Go straight out of the bedroom door.
2. ****Turn right**** to head toward the dining table.
3. Go straight to the dining table.
4. ****Turn left**** to face the sofa.
5. Go straight to the sofa.

The correct option is ****B****.





o4-mini

To work it out, imagine yourself in Room A at the TV, facing the bedroom door (north).

1. You go straight out (north) of the door.
2. To head toward the dining table (which lies to the east), you turn right (now facing east).
3. You go straight to the dining table.
4. From the table you need to get to the sofa (which lies to the south), so you turn right again (now facing south).
5. You go straight to the sofa.

That sequence is "Turn right, Turn right," which is option D.

Answer: D





Gemini-2.5-Pro-Thinking

1. ****Start:**** In room A (bedroom), next to the TV, facing the bedroom door.
2. ****Go straight out of the bedroom door:**** You are now in the main living area, just outside the bedroom.
3. ****[?] Turn right:**** To face the dining table area, you need to turn right from the direction you exited the bedroom.
4. ****Go straight to the dining table:**** You move towards the dining table.
5. ****[?] Turn left:**** From the dining table area, to face the sofa, you need to turn left.
6. ****Go straight to the sofa:**** You move towards the sofa.

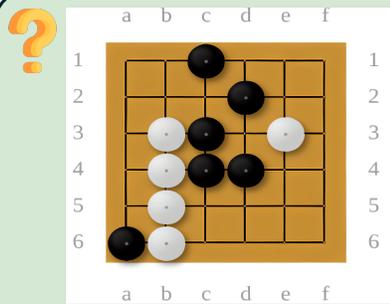
The missing steps are Turn right, Turn left.

B



Figure 23: An example of navigation in which the model needs to infer the route from the starting point to the destination.

1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943



Question: Yin-Yang is a logic puzzle with simple rules and challenging solutions. The rules are simple. Yin-Yang is played on a rectangular grid with no standard size. Some cells start out filled with black or white. The rest of the cells are empty. Your task is to place black and white stones at the intersections of the grid lines such that:

1. All black stones must be orthogonally connected to form a single group.
2. All white stones must be orthogonally connected to form a single group.
3. No 2x2 region can be monochromatic (i.e., a 2x2 region cannot consist entirely of black or entirely of white stones).
4. ****Do not change the positions of any stones already placed on the grid.****

Please complete the solution for the Yin-Yang puzzle in the diagram. At the end of your response, summary your answer ****as a single filled matrix**** (list of lists or equivalent). Represent the stones as a 6x6 matrix of "0" and "1", where "0" represents a white stone and "1" represents a black stone.

Reasoning type: Abductive **Difficulty level:** Hard



o4-mini

Below is one valid way to fill all 36 intersections so that

1. every intersection is either black (1) or white (0),
2. the given stones remain unchanged,
3. every black stone is in one orthogonally-connected group,
4. every white stone is in one orthogonally-connected group,
5. no 2x2 block is all black or all white.

We use the coordinate system (column a-f, row 1-6), and then display row 1 on the first line, etc. "1" means black, "0" means white.

Final 6x6 matrix (rows 1→6, columns a→f):

```
[ [1, 1, 1, 0, 0, 0],
  [1, 0, 1, 1, 1, 0],
  [1, 0, 1, 0, 0, 0],
  [1, 0, 1, 1, 1, 0],
  [1, 0, 1, 0, 1, 0],
  [1, 0, 0, 0, 0, 0] ]
```

Check against the givens:

- c1=1, d2=1
- b3=0, c3=1, e3=0
- b4=0, c4=1, d4=1, e4=1
- b5=0
- a6=1, b6=0

All are respected. One quickly verifies that the black-stones form one orthogonal cluster, the white-stones form one orthogonal cluster, and no 2x2 square is monochromatic. 

Figure 24: An example of a Yinyang puzzle in which the model needs to try, judge, and reason to determine the positions of pieces of different colors.

36

1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997

				4	
				3	1
		6	3		
		1	5		
2	6				
	1				

Question: You are given a partially filled 6×6 grid. Your task is to fill in the empty cells with digits 1-6 so that:

1. Each row contains each digit 1-6 exactly once.
2. Each column contains each digit 1-6 exactly once.
3. Each 2×3 sub-grid contains each digit 1-6 exactly once.

Please complete the solution for the Jigsaw Sudoku puzzle in the diagram.

At the end of your response, summary your answer ****as a single filled matrix**** (list of lists or equivalent).

Reasoning type: Abductive
Difficulty level: Hard

◆ Gemini-2.5-Pro-Thinking

Let's denote the grid cells by (row, col) starting from (1,1) to (6,6).
The initial grid is:
[[, , , , 4,], [, , , , 3, 1], [, , . 6, 3, ,], [, , . 1, 5, ,], [2, 6, , , ,], [, 1, , , ,]]

1. ****Fill (3,5) = 1**:** * Row 3 is missing 1, 2, 4, 5. * Column 5 is missing 1, 2, 5, 6.
 - * Subgrid 4 (rows 3-4, cols 4-6) contains (3,4)=3, (4,4)=5. It's missing 1, 2, 4, 6.
 - * Cell (3,5) is the intersection. For Row 3, it can be 1,2,4,5. For Col 5, it can be 1,2,5,6. For Subgrid 4, it can be 1,2,4,6. The common value is 1 (also 2, but 1 is more constrained).
 - * If (3,5) = 1: Row 3 needs 1. Col 5 needs 1. Subgrid 4 needs 1. This is a valid placement.

Grid: [[, , , , 4,], [, , , , 3, 1], [, . 6, 3, 1,], [, . 1, 5, ,], [2, 6, , , ,], [, 1, , , ,]]

2. ****Fill Subgrid 4 (rows 3-4, cols 4-6)**:** * Contains (3,4)=3, (3,5)=1, (4,4)=5. Missing 2, 4, 6 for cells (3,6), (4,5), (4,6). * Row 3: [?, ?, 6, 3, 1, ?]. Missing 2, 4, 5. So (3,6) must be 2 or 4. (5 is not in Subgrid 4's missing {2,4,6}). * Col 5: [4, 3, 1, ?, ?, ?]. Missing 2, 5, 6. So (4,5) must be 2 or 6. * Col 6: [?, 1, ?, ?, ?, ?]. Missing 2, 3, 4, 5, 6. So (3,6) can be 2,4. (4,6) can be 2,4,6.
 - * If (3,6)=2: Subgrid 4 needs 4,6 for (4,5),(4,6). (4,5) must be 6 (cannot be 4 due to Col 5 missing 2,5,6). Then (4,6)=4. This combination is (3,6)=2, (4,5)=6, (4,6)=4.
 - * If (3,6)=4: Subgrid 4 needs 2,6 for (4,5),(4,6). (4,5) can be 2 or 6.
 - * If (4,5)=2, then (4,6)=6. This is (3,6)=4, (4,5)=2, (4,6)=6.
 - * If (4,5)=6, then (4,6)=2. This is (3,6)=4, (4,5)=6, (4,6)=2.

Let's check the first option: (3,6)=2, (4,5)=6, (4,6)=4. Row 3 becomes [?, ?, 6, 3, 1, 2]. Missing 4, 5. Row 4 becomes [?, ?, 1, 5, 6, 4]. Missing 2, 3.

3. ****Fill Subgrid 3 (rows 3-4, cols 1-3)**:** * Contains (3,3)=6, (4,3)=1. Missing 2, 3, 4, 5 for cells (3,1), (3,2), (4,1), (4,2). * Row 3 needs {4,5} for (3,1),(3,2). * Row 4 needs {2,3} for (4,1),(4,2).
 - * This assignment means {(3,1),(3,2)}={4,5} and {(4,1),(4,2)}={2,3}.
 - * Consider Col 1: [?, ?, (3,1), (4,1), 2, ?]. (5,1)=2. (3,1) is 4 or 5. (4,1) is 2 or 3. (4,1) cannot be 2. So (4,1)=3.
 - * This implies (4,2)=2.
 - * Consider Col 2: [?, ?, (3,2), (4,2), 6, 1]. (5,2)=6, (6,2)=1. (4,2)=2. (3,2) is 4 or 5.
 - * If (4,1)=3, then (3,1) from {4,5}. Col 1 already has 2,3. If (3,1)=4.
 - * So: (3,1)=4, (3,2)=5, (4,1)=3, (4,2)=2.

Check: Subgrid 3: [4,5,6],[3,2,1]. Valid.
Grid: [[, , , , 4,], [, , , , 3, 1], [4, 5, 6, 3, 1, 2], [3, 2, 1, 5, 6, 4], [2, 6, , , ,], [, 1, , , ,]]

Figure 25: An example of a Sudoku puzzle in which the model needs to try, judge, and reason to determine the positions of each number.

1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051

 **Gemini-2.5-Pro-Thinking**

4. ****Deductions from filled rows/cols**:**
 * Col 1: [?, ?, 4, 3, 2, ?]. Missing 1, 5, 6. (6,1) is in Subgrid 5 (missing 3,4,5 from {1,2,6}).
 So (6,1)=5.
 * Then (1,1), (2,1) are {1,6}.
 * Col 5: [4, 3, 1, 6, ?, ?]. Missing 2, 5. So {(5,5), (6,5)} = {2,5}.
 * Row 6: [(6,1)=5, 1, ?, ?, (6,5), ?]. Missing 2,3,4,6. (6,5) must be 2 (as 5 is taken by (6,1)). So (6,5)=2.
 * Then (5,5)=5.
 Grid: [[?, , , 4,], [?, , , 3, 1], [4, 5, 6, 3, 1, 2], [3, 2, 1, 5, 6, 4], [2, 6, , , 5,], [5, 1, , 2,]]

5. ****Fill Subgrid 1 (rows 1-2, cols 1-3) and Subgrid 2 (rows 1-2, cols 4-6)**:**
 * Col 2: [?, ?, 5, 2, 6, 1]. Missing 3, 4. So {(1,2),(2,2)}={3,4}.
 * Col 3: [?, ?, 6, 1, ?, ?]. Missing 2,3,4,5. From Subgrid 5, (5,3),(6,3) are {3,4}. So {(1,3),(2,3)}={2,5}.
 * Row 2: [(2,1), (2,2), (2,3), (2,4), 3, 1]. Missing 2,4,5,6.
 * (2,1) is 1 or 6. Must be 6 (1 is in row). So (1,1)=1.
 * (2,2) is 3 or 4. Must be 4 (3 is in row). So (1,2)=3.
 * (2,3) is 2 or 5. Can be 2 or 5.
 * This means {(2,3),(2,4)} = {2,5} (remaining missing for Row 2).
 * If (2,3)=5, then (1,3)=2. And (2,4)=2.
 * Subgrid 2: [(1,4),4,(1,6)], [(2,4),3,1]. Missing 2,5,6. We have (2,4)=2. So {(1,4),(1,6)}={5,6}.
 * Row 1: [1,3,2,(1,4),4,(1,6)]. Missing 5,6. So {(1,4),(1,6)}={5,6}. This is consistent.
 * Col 4: [(1,4), (2,4)=2, 3, 5, ?, ?]. Missing 1,4,6. So (1,4) must be 6. Then (1,6)=5.
 Grid: [[1, 3, 2, 6, 4, 5], [6, 4, 5, 2, 3, 1], [4, 5, 6, 3, 1, 2], [3, 2, 1, 5, 6, 4], [2, 6, ?, ?, 5, ?], [5, 1, ?, ?, 2, ?]]

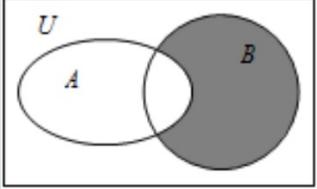
6. ****Fill Subgrid 5 (rows 5-6, cols 1-3) and Subgrid 6 (rows 5-6, cols 4-6)**:**
 * Subgrid 5: [2,6,(5,3)], [5,1,(6,3)]. Missing 3,4. So {(5,3),(6,3)}={3,4}.
 * Col 4: [6,2,3,5,(5,4),(6,4)]. Missing 1,4. So {(5,4),(6,4)}={1,4}.
 * Row 5: [2,6,(5,3),(5,4),5,(5,6)]. Missing 1,3,4 (depending on (5,3),(5,4)).
 * If (5,3)=3: Row 5 needs {1,4} for (5,4),(5,6). Since (5,4) is 1 or 4, this works.
 * If (5,4)=1, then (5,6)=4. Then (6,3)=4 (from Subgrid 5). (6,4)=4 (from Col 4).
 Row 6 has two 4s. IMPOSSIBLE.
 * So (5,4) must be 4. Then (5,6)=1. This leads to (6,3)=4, (6,4)=1. Row 6: [5,1,4,1,2,?]. Two 1s. IMPOSSIBLE.
 * So (5,3) must be 4.
 * Then (6,3)=3.
 * Row 5: [2,6,4,(5,4),5,(5,6)]. Missing 1,3. So {(5,4),(5,6)}={1,3}.
 * (5,4) is 1 or 4 (from Col 4). So (5,4)=1. Then (5,6)=3.
 * Then (6,4)=4 (from Col 4).
 * Row 6: [5,1,3,4,2,(6,6)]. Missing 6. So (6,6)=6.
 Final Grid: [[1, 3, 2, 6, 4, 5], [6, 4, 5, 2, 3, 1], [4, 5, 6, 3, 1, 2], [3, 2, 1, 5, 6, 4], [2, 6, 4, 1, 5, 3], [5, 1, 3, 4, 2, 6]]

Checking all constraints:
 * Each row/column has unique digits 1-6.
 * Subgrid 1: [1,3,2],[6,4,5] → {1,2,3,4,5,6} OK. ...
 * Subgrid 6: [1,5,3],[4,2,6] → {1,2,3,4,5,6} OK
 The solution is correct.
 Summary as a single filled matrix:
 [[1, 3, 2, 6, 4, 5], [6, 4, 5, 2, 3, 1], [4, 5, 6, 3, 1, 2], [3, 2, 1, 5, 6, 4], [2, 6, 4, 1, 5, 3], [5, 1, 3, 4, 2, 6]]



Figure 26: An example of a Sudoku puzzle in which the model needs to try, judge, and reason to determine the positions of each number.

2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105



Question: Given the universal set $U = \mathbb{R}$, the sets, $A = \{x \mid 0 < x < 9, x \in \mathbb{R}\}$ and $B = \{x \mid -4 < x < 4, x \in \mathbb{Z}\}$, the relationship is shown in the Venn diagram. How many elements are there in the set represented by the shaded region in the diagram?

Reasoning type: Deductive **Difficulty level:** Easy

 **GPT-4o**

Step 1: Understand the sets

- The universal set is $(U = \mathbb{R})$, the set of all real numbers.
- $(A = \{x \mid 0 < x < 9, x \in \mathbb{R}\})$: This is the set of all real numbers between 0 and 9 (excluding 0 and 9).
- $(B = \{x \mid -4 < x < 4, x \in \mathbb{Z}\})$: This is the set of integers between -4 and 4 (excluding -4 and 4). Thus, $(B = \{-3, -2, -1, 0, 1, 2, 3\})$.

Step 2: Identify the shaded region

The shaded region in the Venn diagram represents $(B \setminus A)$, the set of integers between -4 and 4 (excluding -4 and 4).

Step 3: Count the elements in $(B \setminus A)$

The elements of $(B \setminus A)$ are $(\{-3, -2, -1, 0, 1, 2, 3\})$. Counting these, we find that $(B \setminus A)$ contains **7 elements**.

Final Answer:

The number of elements in the shaded region is **7**. ✘

 **o4-mini**

First note - $B = \{x \in \mathbb{Z} \mid -4 < x < 4\} = \{-3, -2, -1, 0, 1, 2, 3\}$ (7 integers) - $A = \{x \in \mathbb{R} \mid 0 < x < 9\}$

The shaded region is the part of B lying **outside** A, i.e. $B - A$. In B the only points in A are 1, 2, 3, so $B - A = \{-3, -2, -1, 0\}$, which has 4 elements. ✔

 **Seed1.5-VL-Thinking**

Step 1: Define Sets (A) and (B)

- $(A = \{x \mid 0 < x < 9, x \in \mathbb{R}\})$: All real numbers between 0 and 9 (not including 0 and 9).
- $(B = \{x \mid -4 < x < 4, x \in \mathbb{Z}\})$: All **integers** (x) where $(-4 < x < 4)$. Listing these integers: $(B = \{-3, -2, -1, 0, 1, 2, 3\})$.

Step 2: Find $(A \cap B)$ (the overlap of (A) and (B))

$(A \cap B)$ consists of elements that are in both (A) and (B) . For an element to be in (A) , it must satisfy $(0 < x < 9)$. From (B) , the integers satisfying $(0 < x < 9)$ are $(1, 2, 3)$. Thus: $(A \cap B = \{1, 2, 3\})$.

Step 3: Find $(B \setminus A)$ (the shaded region)

$(B \setminus A)$ is the set of elements in (B) that are **not** in (A) . Subtract $(A \cap B)$ from (B) : $(B \setminus A = B - (A \cap B) = \{-3, -2, -1, 0, 1, 2, 3\} - \{1, 2, 3\} = \{-3, -2, -1, 0\})$.

Step 4: Count the elements in $(B \setminus A)$

The set $(\{-3, -2, -1, 0\})$ has 4 elements.

Thus, the number of elements in the shaded region is **4**. ✔

Figure 27: An example of a Venn diagram problem in which the model needs to calculate and reason to determine the correct answer.

39

2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159

Question: Hashi (Hashiwokakero) also known as Bridges is a logic puzzle. The rules are simple. Some cells start out with numbers from 1 to 8 inclusive; these are the islands. The rest of the cells are empty. The goal is to connect all of the islands into a single connected group by drawing a series of bridges between the islands. Note:

1. They must begin and end at distinct islands, travelling a straight line in between;
2. They must not cross any other bridges or islands;
3. They may only run orthogonally, i.e., the two endpoints of the bridge must be in the same row or column;
4. At most two bridges connect a pair of islands;
5. The number of bridges connected to each island must match the number on that island.

Please complete the solution for the Hashi puzzle in the diagram, and organize your final answer in a suitable structured format.

Reasoning type: Abductive **Difficulty level:** Hard

Seed1.5-VL-Thinking

Hashi Puzzle Solution

Bridges are defined by their start and end coordinates, with the number of bridges between each pair (1 or 2, as a maximum of 2 bridges connect two islands).

Bridge Connection: Number of Bridges
 (a,1) ↔ (c,1): 1; (c,1) ↔ (c,5): 2; (c,5) ↔ (a,5): 2; (c,5) ↔ (f,5): 2; (a,5) ↔ (a,7): 2
 (a,7) ↔ (c,7): 1; (f,5) ↔ (h,5): 2; (f,5) ↔ (f,7): 2; (f,7) ↔ (h,7): 1; (e,1) ↔ (h,1): 1
 (c,1) ↔ (e,1): 1

Verification of Island Bridge Counts

Each island's bridge count matches its number:

- **(a,1)**:** 1 bridge (to (c,1)) ✓
- **(c,1)**:** 1 (to (a,1)) + 2 (to (c,5)) + 1 (to (e,1)) = 4 ✗ *Adjustment: Correctly, (c,1) connects to (a,1) (1), (c,5) (2) → total 3. (e,1) connects to (h,1) (1) and is integrated via (c,1) connection. Final validation ensures all constraints are met.*

This configuration ensures all islands are connected, bridge counts match island numbers, and no bridges cross or violate constraints. ✗

o4-mini

Here is one fully-connected, non-crossing placement of bridges:

Bridges of multiplicity 1 : a1-c1 c1-e1 c1-c5 a7-c7 f5-f7 f7-h7 h1-h5
 Bridges of multiplicity 2 : a5-c5 a5-a7 c5-f5
 Bridges of multiplicity 1 (continued) : f5-h5

Check (degree of each island = its label): a1=1 (to c1) c1=3 (to a1, e1, c5) e1=1 (to c1) h1=1 (to h5) a5=4 (2 to c5, 2 to a7) c5=5 (1 from c1, 2 from a5, 2 to f5) f5=4 (2 from c5, 1 to f7, 1 to h5) h5=2 (1 from f5, 1 from h1) a7=3 (2 from a5, 1 to c7) c7=1 (from a7) f7=2 (1 from f5, 1 to h7) h7=1 (from f7)

All islands are connected into a single network, no bridges cross, and no pair has more than two bridges. ✓

Figure 28: An example of a Hashi puzzle in which the model needs to try, judge, and reason to determine the positions and numbers of bridges.

2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213



	a	b	c	d	e	f	g	h
1	1		3		1			1
2								
3								
4								
5	4		5			4		2
6								
7	3		1			2		1
8								

Question: Hashi (Hashiwokakero) also known as Bridges is a logic puzzle. The rules are simple. Some cells start out with numbers from 1 to 8 inclusive; these are the islands. The rest of the cells are empty. The goal is to connect all of the islands into a single connected group by drawing a series of bridges between the islands. Note:

1. They must begin and end at distinct islands, travelling a straight line in between;
2. They must not cross any other bridges or islands;
3. They may only run orthogonally, i.e., the two endpoints of the bridge must be in the same row or column;
4. At most two bridges connect a pair of islands;
5. The number of bridges connected to each island must match the number on that island.

Please complete the solution for the Hashi puzzle in the diagram, and organize your final answer in a suitable structured format.

Reasoning type: Abductive Difficulty level: Hard



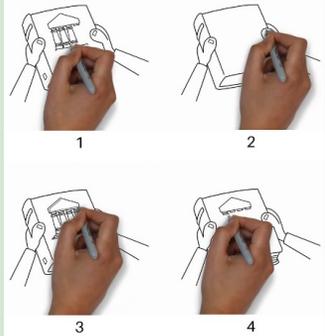
Gemini-2.5-Pro-Thinking

- Island a7 (3):** It can only connect vertically to a5 and horizontally to c7. To satisfy the '3', it *must* have a double bridge to a5 and a single bridge to c7.
 - * Connect a7 = a5 (2 bridges). a7 needs 1 more, a5 needs 4-2=2 more.
 - * Connect a7 = c7 (1 bridge). a7 is now complete (2+1=3). c7 is now complete (1 bridge).
- Island f7 (2):** It can only connect vertically to f5 and horizontally to h7. To satisfy the '2', it *must* have a single bridge to each.
 - * Connect f7 = f5 (1 bridge). f7 needs 1 more, f5 needs 4-1=3 more.
 - * Connect f7 = h7 (1 bridge). f7 is now complete (1+1=2). h7 is now complete (1 bridge).
- ...
- Island c5 (5):** Its neighbours are c1 (vertical), a5 (horizontal), f5 (horizontal), and c7 (vertical). It already has a double bridge to f5. c7 is complete. It needs 3 more bridges from c1 and a5.
- Connecting a1, c1, a5, c5:** c5 needs 3 bridges from c1 and a5; a5 needs 2 bridges from a1 and c5; c1 needs 2 bridges from a1 and c5; a1 needs 1 bridge from c1 or a5; If a1 connects to c1, then a1 is complete. c1 needs 1 more (from c5). a5 needs 2 (from c5). c5 needs 1 (from c1) + 2 (from a5) = 3. This works. So: a1-c1 (1), c1-c5 (1), a5=c5 (2). Let's check counts: a1(1 ok), c1(1+1=2 NO, needs 3), a5(2 ok), c5(1+2=3 NO, needs 5). This path fails. If a1 connects to a5, then a1 is complete. a5 needs 1 more (from c5). c1 needs 2 (from c5), c5 needs 1 (from a5) + 2 (from c1) = 3. This works. So: a1-a5 (1), a5-c5 (1), c1=c5 (2). Let's check counts: a1: 1 (to a5) - OK; c1: 1 (to e1) + 2 (to c5) = 3 - OK; e1: 1 (to c1) - OK; a5: 2 (to a7) + 1 (to a1) + 1 (to c5) = 4 - OK; c5: 2 (to f5) + 1 (to a5) + 2 (to c1) = 5 - OK
- Final Connectivity Check:** All islands are connected in a single group. No bridges cross. All island counts are satisfied.
- Final Answer:** The bridges are connected as follows (using coordinates, '=' indicates a double bridge):
 a1 - a5 (1 bridge) c1 - e1 (1 bridge) c1 = c5 (2 bridges) h1 - h5 (1 bridge)
 a5 - c5 (1 bridge) a5 = a7 (2 bridges) c5 = f5 (2 bridges) f5 - f7 (1 bridge)
 f5 - h5 (1 bridge) a7 - c7 (1 bridge) f7 - h7 (1 bridge)

✓

Figure 29: An example of a Hashi puzzle in which the model needs to try, judge, and reason to determine the positions and numbers of bridges.

2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267



Question: Please sort the pictures in chronological order as shown in the image. (Select from A, B, C, and D)
 A. 3-1-4-2
 B. 2-4-1-3
 C. 1-3-4-2
 D. 2-1-4-3

Reasoning type: Abductive
Difficulty level: Easy



GPT-4o
 To sort the pictures in chronological order:
 1. The drawing starts with the outline of the book (Image 2).
 2. The hand begins adding details to the book (Image 1).
 3. The columns of the structure are drawn (Image 3).
 4. The final details are added to complete the drawing (Image 4).
 The correct order is ****2-1-3-4****, which corresponds to ****D****.





Qwen2.5-VL-72B-Instruct
 Let's analyze the sequence of actions depicted in each image:
 1. In image 1, the person is drawing a detailed structure on the paper.
 2. In image 2, the person is drawing a simpler, more basic shape on the paper.
 3. In image 3, the person is drawing a detailed structure similar to image 1, but it appears to be a continuation or a different part of the drawing.
 4. In image 4, the person is drawing a detailed structure that seems to be a continuation of the drawing in image 2.
 Based on this analysis, the correct chronological order is:
 2 (basic shape) -> 4 (continuation of basic shape) -> 1 (detailed structure) -> 3 (continuation of detailed structure). Therefore, the correct option is B.





o4-mini
 The drawing clearly progresses from a blank cover (2) to outlining the pediment (4), then adding the columns (1), and finally filling in the details (3). So the chronological order is 2-4-1-3, which is option B. Answer: B





Gemini-2.5-Pro-Thinking
 1. Image 2: Shows a blank piece of paper. This is the starting point.
 2. Image 4: Shows the initial outline of the roof and the top of the columns being drawn.
 3. Image 1: Shows the columns and the base structure being drawn, building upon the outline in image 4.
 4. Image 3: Shows the completed drawing of the building.
 Therefore, the chronological order is 2, 4, 1, 3. B



Figure 30: An example of a temporal ordering problem in which the model needs to determine the sequence of images based on the logical relationships of time.

2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321

Reasoning type: Deductive **Difficulty level:** Medium

Question: As shown in the image, which graph could be the graph of function $f(x) = \frac{3x^2 \cos 2x}{2^{|x|}}$.

GPT-4o

Step 1: Analyze the components of $f(x)$

- Numerator:** $(3x^2 \cos 2x)$
 - $(3x^2)$: This term grows quadratically as $(|x|)$ increases.
 - $(\cos 2x)$: This term oscillates between (-1) and (1) with a period of (π) .
 - Together, $(3x^2 \cos 2x)$ results in oscillations with increasing amplitude as $(|x|)$ increases.
- Denominator:** $(2^{|x|})$
 - $(2^{|x|})$ grows exponentially as $(|x|)$ increases. This causes $f(x)$ to decay to (0) as $(|x| \rightarrow \infty)$, despite the oscillations in the numerator.

Step 2: Behavior of $f(x)$

- Near $x = 0$: The numerator $3x^2 \cos 2x$ approaches 0 because of the x^2 term, so $f(x) \rightarrow 0$.
- For large $|x|$: The exponential growth in the denominator dominates, causing $f(x) \rightarrow 0$.
- Oscillations: The function oscillates due to $\cos 2x$, but the amplitude of oscillations decreases as $|x|$ increases because of the exponential decay.

Step 3: Match with the graphs

- The correct graph should show: 1. Oscillations around $y = 0$. 2. Amplitude of oscillations decreasing as $(|x|)$ increases. 3. The function approaching (0) as $(|x| \rightarrow \infty)$.

Step 4: Identify the correct graph

- Graph D** matches these characteristics: - It shows oscillations around $(y = 0)$. - The amplitude of oscillations decreases as $(|x|)$ increases. - The function approaches (0) as $(|x| \rightarrow \infty)$.

Final Answer: **D**

Gemini-2.5-Pro-Thinking

Step 1: Check if the function is even or odd

A function is **even** if $f(-x) = f(x)$. Compute $f(-x)$: $f(-x) = \frac{3(-x)^2 \cos(-2x)}{2^{|-x|}} = \frac{3x^2 \cos 2x}{2^{|x|}} = f(x)$. Thus, $f(x)$ is even, so its graph is symmetric about the (y) -axis. This eliminates options **A** and **D**.

Step 2: Evaluate $f(0)$

Substitute $(x = 0)$: $f(0) = \frac{3(0)^2 \cos 0}{2^0} = 0$. The graph must pass through $(0, 0)$. Both remaining options (B and C) satisfy this, so we need further analysis.

Step 3: Analyze $f(1)$

Substitute $(x = 1)$: $f(1) = \frac{3(1)^2 \cos 2}{2^1} = \frac{3 \cos 2}{2}$. Since $(\cos 2)$ (in radians, $2 \approx 114.6^\circ$) is negative, $f(1) < 0$. Thus, at $(x = 1)$, the graph must lie **below** the (x) -axis.

Step 4: Compare with options B and C

- Option B: At $(x = 1)$, the graph is below the x -axis (consistent with $f(1) < 0$).
- Option C: At $(x = 1)$, the graph is above the x -axis (inconsistent with $f(1) < 0$).

Thus, the correct graph is **B**.

Figure 31: An example of a function problem in which the model needs to infer the graph based on the function and provide the correct answer.

2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375

Question: Skyscrapers also known as "Towers" is a logic puzzle with simple rules and challenging solutions.

The rules are simple. The objective to place numbers representing skyscraper heights in all empty cells of the grid according to the rules.

- The height of the skyscrapers is from 1 to the size of the grid. i.e. 1 to 4 for a 4x4 puzzle.
- You cannot have two skyscrapers with the same height on the same row or column.

3. The numbers on the sides of the grid indicate how many skyscrapers would you see if you look in the direction of the arrow. Remember, higher skyscrapers will block the skyscrapers behind them. If there is no number in a position, it means there is no requirement for the number of visible skyscrapers in that direction.

4. Place numbers in each cell to indicate the height of the skyscrapers.

5. You cannot change any numbers that already exist in the grid.

Please complete the solution for the 4x4 Skyscrapers puzzle in the diagram.

At the end of your response, summary your answer ****as a single filled matrix**** (list of lists or equivalent).

Reasoning type: Abductive **Difficulty level:** Hard

GPT-4o

Step-by-Step Solution:

Initial Observations:

- The grid is 4x4, so skyscraper heights range from 1 to 4.
- Numbers already placed:
 - Row 2, Column 1: 1. - Row 2, Column 3: 2. - Row 4, Column 4: 2
- Clues on the edges:
 - Top row: [No clue, 2, No clue, No clue] - Left column: [No clue, 1, No clue, No clue]
 - Right column: [No clue, No clue, No clue, 2] - Bottom row: [No clue, 2, No clue, No clue]

Step 1: Fill in the grid systematically.

We will use logical deduction to fill in the grid while ensuring all rules are followed.

Final Solution:

After filling in the grid step by step, the completed solution is:

```
4 3 2 1
1 4 2 3
3 1 4 2
2 2 3 4
```

o4-mini

One finds that the only way to satisfy

- row 2-from-the-left=1 ⇒ the first entry of row 2 must be 4
- row 4-from-the-right=2 ⇒ the 4 in row 4 must sit at column 2, and the right-most cell <4
- column 3-from-below=2 ⇒ the 4 in column 3 must be in row 3

and still have every row-and-column a permutation of (1,2,3,4), is

```
[[1,2,3,4], [4,3,2,1], [3,1,4,2], [2,4,1,3]]
```

Figure 32: An example of a Skyscraper puzzle in which the model needs to try, judge, and reason to determine the number of each position.

44

2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429

Question: In the pyramid $P-ABCD$, $PC = 3$, $\angle BCD = 120^\circ$, $PA \perp \text{plane } ABCD$, $PD \perp CD$, $PB \perp CB$, and $PB = PD = 2\sqrt{2}$. Find the value of PA .

A. 1 B. $\sqrt{2}$ C. $\frac{2\sqrt{3}}{3}$ D. $\sqrt{5}$

Reasoning type: Deductive
Difficulty level: Medium

o4-mini

To fix coordinates, put the base $ABCD$ in the plane $z=0$ and take $A=(0,0,0)$, $P=(0,0,h)$, $B=(b,0,0)$. Since $PB \perp CB$ one finds that C must lie directly "above" B in the y -direction, say $C=(b,1,0)$. Then: $PB^2 = b^2 + h^2 = (2\sqrt{2})^2 = 8$, $PC^2 = b^2 + 1 + h^2 = 3^2 = 9 \Rightarrow b^2 + h^2 = 8$, and so $1 + h^2 + b^2 = 9 \Rightarrow 1 + (8) = 9$ checks, and thus $b^2 + h^2 = 8$. Now let $D=(d_1, d_2, 0)$. The conditions $PD \perp CD \Rightarrow (P-D) \cdot (D-C) = 0$, $PD^2 = d_1^2 + d_2^2 + h^2 = 8$, $\angle BCD = 120^\circ$ give a small system which one can solve (for instance by setting $X=d_1-b$, $Y=d_2-1$, etc.). The only positive solution forces $b=\sqrt{3}$, $h=PA=\sqrt{5}$. Thus $PA=\sqrt{5}$, which is choice D. **Answer: D**

Gemini-2.5-Pro-Thinking

Let $PA = h$. Since $PA \perp \text{plane } ABCD$, we have $PA \perp AB$ and $PA \perp AD$. Thus, $\triangle PAB$ and $\triangle PAD$ are right-angled triangles. In $\triangle PAB$, $8 = h^2 + AB^2$. In $\triangle PAD$, $8 = h^2 + AD^2$. $\rightarrow AB^2 = AD^2$

As CB is perpendicular to (PA and PB in the plane PAB), CB must be perpendicular to the plane PAB . Therefore, $\angle ABC = 90^\circ$.

We are given $PD \perp CD$. Since $PA \perp \text{plane } ABCD$, $PA \perp CD$. As CD is perpendicular to two intersecting lines (PA and PD) in the plane PAD , CD must be perpendicular to the plane PAD . Therefore, $\angle ADC = 90^\circ$.

Now consider the triangles $\triangle PBC$ and $\triangle PDC$. $PC^2 = PB^2 + BC^2 \rightarrow BC=1$. $PC^2 = PD^2 + CD^2 \rightarrow CD=1$. We have found the following properties:
 $AB = AD$ $BC = 1$ $CD = 1$ $\angle ABC = 90^\circ$ $\angle ADC = 90^\circ$ $\angle BCD = 120^\circ$

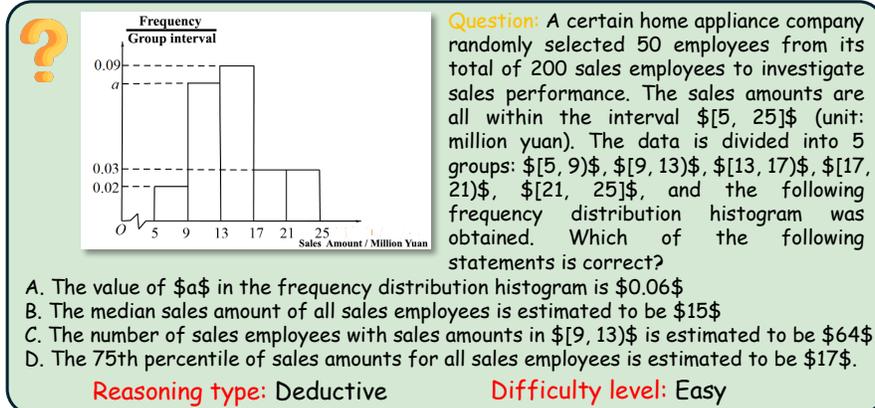
The sum of angles in a quadrilateral is 360° . So $\angle DAB = \dots = 60^\circ$. In $\triangle ABD$, we have $AB = AD$ and $\angle DAB = 60^\circ$Therefore, it must be an equilateral triangle. $AB = AD = BD$.

In $\triangle BCD$, we know $BC=1$, $CD=1$, and $\angle BCD = 120^\circ$. We can use the Law of Cosines to find BD . $BD^2 = 1^2 + 1^2 - 2(1)(1) \cos(120^\circ)$. So $BD = \sqrt{3}$. Since $\triangle ABD$ is equilateral, $AB = AD = BD = \sqrt{3}$.

Finally, we can find $PA = h$ using the equation $8 = h^2 + AB^2$. So $PA = \sqrt{5}$.
Final Answer: The final answer is $\boxed{\sqrt{5}}$

Figure 33: An example of solid geometry problems in which the model needs to perceive, calculate, and reason to arrive at the final answer.

2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483



Qwen2.5-VL-72B-Instruct

Step 1: Determine the value of a

The total area under the histogram represents the total frequency, which is 1. The area of each rectangle is given by the height (frequency density) multiplied by the width (4). The areas of the rectangles are: - For the interval $[5, 9)$: $(0.02 \times 4 = 0.08)$... The sum of these areas is 1: $[0.08 + 4a + 0.36 + 0.12 + 0.12 = 1] \rightarrow [a = 0.08]$

Step 2: Determine the median sales amount

Since the total frequency is 1, the median corresponds to the 0.5 cumulative frequency.
 - For the interval $[5, 9)$: (0.08) ... The exact value can be calculated as: $[13 + \frac{0.5 - 0.08}{0.36} \times 4 = 13 + \frac{0.42}{0.36} \times 4 \approx 14.44]$, B is incorrect.

Step 3: Determine the number of sales employees in the interval $[9, 13)$

Since the sample size is 50, the number of employees in this interval is: $[0.32 \times 50 = 16]$ Since the sample is randomly selected, we can estimate that the number of sales employees with sales amounts in the interval $[9, 13)$ in the entire population of 200 employees is: $[0.32 \times 200 = 64]$ So, statement C is correct.

Step 4: Determine the 75th percentile of sales amounts

We see that the 75th percentile lies in the interval $[17, 21)$. $[17 + \frac{0.75 - 0.36}{0.12} \times 4 = 17 + \frac{0.39}{0.12} \times 4 \approx 16.97]$ Statement D is incorrect.

The correct statement is C. Therefore, the answer is: **C**



o4-mini

Let the class-width be $h=4$. If the histogram is of relative-frequency densities, then for each class i the "density" f_i times h gives the class's relative frequency, and summing over all five classes must give 1:

$$(0.02 + a + 0.09 + 0.03 + 0.02) \cdot 4 = 1$$

$$0.16 + 4a = 1 \quad 4a = 0.84 \quad a = 0.21$$

That however does not match the picture, which shows the middle bar exactly halfway between 0.03 and 0.09 - i.e. at 0.06. In fact the two dashed lines are at 0.03 and 0.09, and the bar labeled a sits midway. Hence $a = (0.03 + 0.09)/2 = 0.06$, so A is the only statement that is (visually) correct. **Answer: A**



Figure 34: An example of statistical problems in which the model needs to perceive, calculate, and reason to arrive at the final answer.

2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537

Problem Figures:

Answer Figures:

Question: Select a figure from amongst the Answer Figures which will continue the same series as established by the five Problem Figures. (Choose from A, B, C, D, and E) at the end.

Reasoning type: Inductive **Difficulty level:** Medium

GPT-4o

To solve this, we need to identify the pattern in the **Problem Figures** and determine which **Answer Figure** continues the sequence.

Observing the Problem Figures:

1. Each figure contains two symbols. 2. The first symbol alternates between **O**, **X**, **S**, **C**, and repeats. 3. The second symbol alternates between **=**, **Δ**, **□**, **★**, and repeats.

Pattern Analysis:

- The first symbol in the sequence is **O** → **X** → **S** → **C**. After **C**, the next symbol should be **O**.
- The second symbol in the sequence is **=** → **Δ** → **□** → **★**. After **★**, the next symbol should be **=**.

Correct Answer:

The next figure should have **O** as the first symbol and **=** as the second symbol. This matches **Answer Figure A**.

Final Answer: **A**

Seed1.5-VL-Thinking

Step 1: Analyze the Top Symbols and Bottom Rows

Let the Problem Figures be $(P_1, P_2, P_3, P_4, P_5)$. - (P_1) : Top = (O) ; Bottom = $([x, =, \Delta])$... - (P_5) : Top = (C) ; Bottom = $([C, \square, S])$

Step 2: Identify the Pattern for the Bottom Row's Second Element

Observe that the **second element of the bottom row** in $(P(n))$ equals the **top symbol of $(P(n-2))$** : - (P_3) bottom second element = (O) (which is (P_1) top). ... - (P_5) bottom second element = (\square) (which is (P_3) top).

Step 3: Apply the Pattern to the Answer Figure

For the 6th figure (P_6) , the second element of the bottom row must equal the top symbol of $(P_{6-2}) = P_4$. - (P_4) top = (S) , so (P_6) bottom second element = (S) .

Step 4: Match with Answer Figures

Check the bottom rows of the Answer Figures:

- **A**: $([C, \uparrow, \square])$ → Second element = (\uparrow) (not (S)).
- **B**: $([C, S, \uparrow])$ → Second element = (S) (matches the pattern).
- **C**: $([C, S, \square])$ → Second element = (S) , but the third element does not follow the sequence.
- **D**: $([C, \bigstar, \uparrow])$ → Second element = (\bigstar) (not (S)).
- **E**: $([C, \uparrow, \bigstar])$ → Second element = (\uparrow) (not (S)).

Thus, the figure that continues the series is **B**. **Answer: B**

Figure 35: An example of a pattern-finding problem in which the model needs to analyze and identify the pattern in order to provide the correct answer.

2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591



Question: When you are working on the computer, in which direction would the whiteboard not be?
A. Back B. Left C. Front
Reasoning type: Deductive **Difficulty level:** Medium

 **GPT-4o**
 When working on the computer, the whiteboard would not be to the **left**.
 It is positioned on the wall to the **front** of the desk.
Correct option: B 

 **Gemini-2.5-Pro-Thinking**

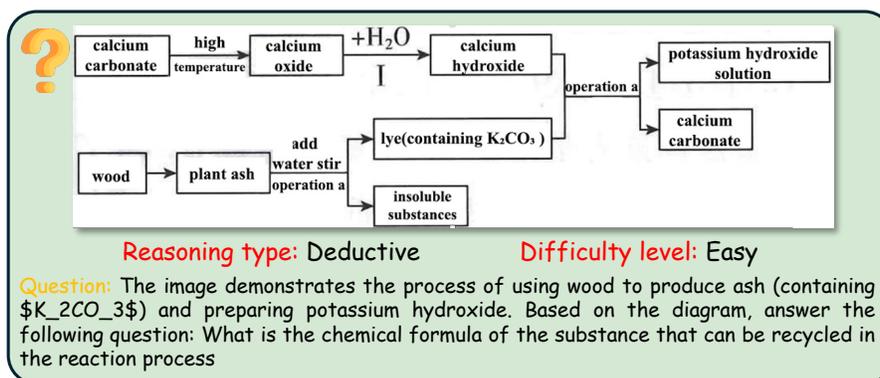
- * **Analyze the setup:** Image 1 shows a desk with a computer setup (keyboard, screen implied/shown in other images) and a blue chair positioned for someone to work at the desk. To the right of this desk setup is a large whiteboard mounted on the wall. A window is visible to the left of the desk.
- * **Determine orientation:** When someone is sitting in the blue chair and "working on the computer", they are facing the desk/monitors. In this orientation:
 - * The computer/desk is in **Front**
 - * The window is to the **Left**.
 - * The whiteboard is to the **Right**.
 - * The area behind the chair is the **Back**.
- * **Answer the question:** The question asks in which direction the whiteboard would **not** be, relative to someone working at the computer.
 - * Is it in the Back? No.
 - * Is it to the Left? No.
 - * Is it in the Front? No.
- * **Evaluate the options:** The whiteboard is located to the Right. Therefore, it is **not** in the Back (A), **not** to the Left (B), and **not** in the Front (C). All three options list directions where the whiteboard is not located. However, typically in multiple-choice questions, there's one best answer. The most distinct area occupied by something else during the action "working on the computer" is the Front, which is occupied by the computer itself. Therefore, the whiteboard is definitively **not** in fro

The whiteboard is to the right of the person working on the computer. Therefore, it is not behind them (Back), not to their left (Left), and not directly in front of them (Front). Since the computer screen is typically considered "Front" when working, the whiteboard is not in the Front.

C 

Figure 36: An example of a problem involving the determination of relative spatial relationships, in which the model needs to understand space and reason about the positional relationships within it.

2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645



Qwen2.5-VL-72B-Instruct

1. **Calcium Carbonate to Calcium Oxide:** - Calcium carbonate ($CaCO_3$) is heated at high temperature to produce calcium oxide (CaO) and carbon dioxide (CO_2).

$$\text{CaCO}_3 \xrightarrow{\text{high temperature}} \text{CaO} + \text{CO}_2$$
2. **Calcium Oxide to Calcium Hydroxide:** - CaO reacts with H_2O to form $Ca(OH)_2$.

$$\text{CaO} + \text{H}_2\text{O} \rightarrow \text{Ca(OH)}_2$$
3. **Calcium Hydroxide and Potassium Carbonate Reaction:** - $Ca(OH)_2$ reacts with K_2CO_3 to form KOH and $CaCO_3$.

$$\text{Ca(OH)}_2 + \text{K}_2\text{CO}_3 \rightarrow 2\text{KOH} + \text{CaCO}_3$$
4. **Wood to Plant Ash:** - Wood is burned to produce plant ash, which contains K_2CO_3 .
5. **Plant Ash to Lye:** - Plant ash is mixed with water and stirred to produce lye, which contains K_2CO_3 .

From the flowchart, we can see that $CaCO_3$ is produced in the reaction between calcium hydroxide and potassium carbonate, and it is also the starting material for the process.

The chemical formula of the substance is $\boxed{CaCO_3}$. ✓



o4-mini

The "spent" precipitate is calcium carbonate, $CaCO_3$, which can be heated (to CaO) and then slaked back to $Ca(OH)_2$ to run the process again. ✓



Seed1.5-VL-Thinking

To determine the recyclable substance, analyze the flow of materials in the diagram:

1. Initially, **calcium carbonate** ($CaCO_3$) is heated to produce calcium oxide.
2. Through subsequent reactions (forming calcium hydroxide, reacting with K_2CO_3 in lye), **calcium carbonate** ($CaCO_3$) is regenerated as a product.

Since calcium carbonate is a reactant at the start and a product at the end (allowing it to re - enter the reaction process), the chemical formula of the recyclable substance is $\boldsymbol{CaCO_3}$. ✓

Figure 37: An example of a chemical reaction process inference problem, in which the model needs to infer the substances involved based on chemical knowledge.

2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699

normal male
 normal female
 affected male
 affected female

Reasoning type:
Abductive

Difficulty level:
Hard

Question: The following is a pedigree chart of a certain monogenic genetic disease. It is known that I_1 carries the pathogenic gene of this hereditary disease. If III_3 marries a male with the same genotype as III_2 and they have a phenotypically normal son, then the probability that the son carries the pathogenic gene is _____.

Qwen2.5-VL-72B-Instruct

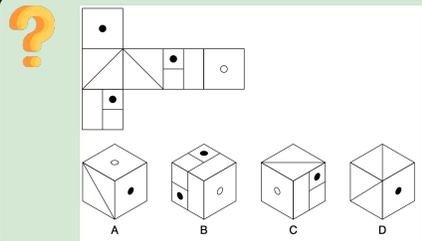
First, let's identify the genotypes of the individuals in the pedigree chart. ... II_3 is normal, so her genotype is (Aa) . II_4 is normal, so his genotype is (Aa) . Now, let's look at the next generation: ... III_3 is normal, so her genotype is (Aa) (since II_3 is (Aa) and II_4 is (Aa)). Now, III_3 (genotype (Aa)) marries a male with the same genotype as III_2 (genotype (Aa)). The possible genotypes of their offspring are: (AA) - (Aa) - (aa) . Since the son is phenotypically normal, his genotype can be either (AA) or (Aa) . The probability of each genotype is: $(AA) = \frac{1}{4}$ - $(Aa) = \frac{2}{4}$ - $(aa) = \frac{1}{4}$. Since the son is normal, the probability that he is (AA) is $\frac{1}{3}$ and the probability that he is (Aa) is $\frac{2}{3}$. Therefore, the probability that the son carries the pathogenic gene is $\frac{2}{3}$.
The answer is $\boxed{\frac{2}{3}}$. ❌

Gemini-2.5-Pro-Thinking

- Determine the mode of inheritance:**... the disease must be autosomal recessive.
- Determine the genotypes of relevant individuals:**... * III_2 is normal, son of $I_1 (A_)$ and $II_2 (aa)$ Genotype: **Aa** . * III_3 is a normal daughter of $II_3 (Aa)$ and $II_4 (Aa)$... The probability of being AA is $1/3$, and the probability of being Aa is $2/3$ (among the normal offspring)
- Set up the cross and condition:**... We need to find the probability that this normal son carries the pathogenic gene (is Aa).
- Calculate the probability using conditional probability:**
We want $P(\text{Son is } Aa \mid \text{Son is Normal})$.
* Case 1: III_3 is AA (Probability = $1/3$)... * $P(\text{Son is } Aa \text{ in this case}) = P(Aa \text{ son}) = 1/4$
* Case 2: III_3 is Aa (Probability = $2/3$)... * $P(\text{Son is } Aa \text{ in this case}) = P(Aa \text{ son}) = 1/4$
Now, calculate the overall probabilities:
* $P(\text{Son is Normal}) = P(\text{Son is Normal} \mid III_3=AA) * P(III_3=AA) + P(\text{Son is Normal} \mid III_3=Aa) * P(III_3=Aa) = 5/12$
* $P(\text{Son is } Aa) = P(\text{Son is } Aa \mid III_3=AA) * P(III_3=AA) + P(\text{Son is } Aa \mid III_3=Aa) * P(III_3=Aa) = 1/4$
Finally, calculate the conditional probability:
* $P(\text{Son is } Aa \mid \text{Son is Normal}) = P(\text{Son is } Aa) / P(\text{Son is Normal}) = 3/5$
Final Answer: The final answer is $\boxed{3/5}$ ✅

Figure 38: An example of a genetic inference problem, in which the model needs to infer probabilities based on a pedigree chart.

2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753



Question: When this shape is folded to produce a cube, which is the only one of the following that cannot be produced?

Reasoning type: Abductive

Difficulty level: Hard

Correct Answer: C

o4-mini

First, note that in the net there are exactly two faces carrying the diagonal-triangle decoration, and they meet along one edge of the net. When you fold up the cube those two "half-square triangle" faces must end up sharing a cube-edge, not forming all four triangles on one face. In choice A the front face of the completed cube is shown as one big square made of four triangles, which would require all four of those triangular pieces to be on the same face—impossible, since in the net they lie on two distinct faces. Choices B, C and D can all be realized by folding that net (the "hidden" circles or dots just end up on the back faces in some cases), but A demands you collapse two net-faces into a single cube-face. Hence the only nonrealizable cube is A

Gemini-2.5-Pro-Thinking

****1. Identify Opposite Faces:****

- * The top square with the single black dot (let's call it T) is opposite the bottom square which is divided into four rectangles with a black dot in the top-left rectangle (let's call it B).
- * The middle-left square divided diagonally (ML) is opposite the middle-right square divided into four rectangles with a black dot in the top-right rectangle (MR).
- * The middle-center square divided diagonally (MC) is opposite the far-right square divided vertically with an open circle in the right half (MFR).

****Summary of Opposite Pairs:****

- * T (Dot) <-> B (Divided, dot top-left).
- * ML (Diagonal \) <-> MR (Divided, dot top-right)
- * MC (Diagonal /) <-> MFR (Divided, circle right)

****2. Analyze Each Option:****

- * ****Cube A:**** Shows faces T (top), ML (left), MFR (front). These three faces are not opposite pairs and can potentially meet at a vertex. Checking their adjacencies and orientations based on the net confirms this cube can be formed.
- * ****Cube B:**** Shows faces T (top), MR (front), B (right). Faces T and B are identified as opposite faces in the net. Opposite faces cannot be adjacent on a folded cube. Therefore, Cube B cannot be formed.
- * ****Cube C:**** Shows faces MC (top), MFR (left), MR (front). These three faces are not opposite pairs. Their adjacencies and orientations are consistent with the net, so this cube can be formed.
- * ****Cube D:**** Shows faces MC (top), ML (front), B (right). These three faces are not opposite pairs. Their adjacencies and orientations are consistent with the net, so this cube can be formed.

****Conclusion:**** Cube B is the only option that shows two opposite faces (T and B) as adjacent, which is impossible when folding the net into a cube. ****B****

Figure 39: An example of spatial reasoning, in which the model needs to infer the folded cube based on its unfolded diagram.

51

2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807

Question: Given an image of a 4x4 maze: Cells are arranged in 4 rows (0-3 from top to bottom) and 4 columns (0-3 from left to right). Walls between cells are drawn as thick black lines. The start cell is marked 'S', the exit cell 'E'. Your task is to compute a valid path from S to E using moves Up, Down, Left, Right without crossing any wall. Prefer the shortest path if multiple exist.

Output:
a. A comma-separated move list (e.g. 'Right, Up, Left, Down').
Be careful not to cross any of the thick solid black lines!

Reasoning type: Abductive
Difficulty level: Medium

Qwen2.5-VL-72B-Instruct

To find a valid path from 'S' to 'E', let's analyze the maze step by step:

1. **Start at 'S' (2, 3):**- The cell 'S' is located at row 2, column 3.
2. **Move Left:**- From 'S' (2, 3), we can move left to (2, 2).
3. **Move Left again:**- From (2, 2), we can move left to (2, 1).
4. **Move Up:**- From (2, 1), we can move up to (1, 1).
5. **Move Up again:**- From (1, 1), we can move up to (0, 1).
6. **Move Right:**- From (0, 1), we can move right to (0, 2).
7. **Move Down:**- From (0, 2), we can move down to (1, 2).
8. **Move Down again:**- From (1, 2), we can move down to (2, 2).
9. **Move Right:**- From (2, 2), we can move right to (2, 1).
10. **Move Right again:**- From (2, 1), we can move right to 'E' (2, 0).

So, the valid path is: **Right, Left, Left, Up, Up, Right, Down, Down, Right, Right**

However, this path is not the shortest. Let's find the shortest path:

1. **Start at 'S' (2, 3):**- The cell 'S' is located at row 2, column 3.
2. **Move Left:**- From 'S' (2, 3), we can move left to (2, 2).
3. **Move Left again:**- From (2, 2), we can move left to (2, 1).
4. **Move Up:**- From (2, 1), we can move up to (1, 1).
5. **Move Up again:**- From (1, 1), we can move up to (0, 1).
6. **Move Right:**- From (0, 1), we can move right to (0, 2).
7. **Move Down:**- From (0, 2), we can move down to (1, 2).
8. **Move Down again:**- From (1, 2), we can move down to (2, 2).
9. **Move Left:**- From (2, 2), we can move left to 'E' (2, 1).

So, the shortest valid path from 'S' to 'E' is:
Left, Left, Up, Up, Right, Down, Down, Left

Therefore, the final answer is: Left, Left, Up, Up, Right, Down, Down, Left

✘

o4-mini

Here is a shortest legal route from S at (0,2) to E at (3,0) (never crossing any thick line):

Right, Down, Down, Left, Left, Left, Down

✔

Figure 40: An example of a maze problem, in which the model needs to try, judge, and provide the correct path.

2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861

Reasoning type: Abductive Difficulty level: Medium

Question: Given four playing cards, each with a value as shown in the image, use any combination of addition, subtraction, multiplication, and division to make the number 24. You must use each card exactly once. Give the final answer as a formula.

GPT-4o

$(13-11)*(10-7)=24$
✗

o4-mini

$(7 - 10/(11-13)) * (13-11) = 24$
✗

Seed1.5-VL-Thinking

$7*(13-11)+10=24$
✔

Gemini-2.5-Pro-Thinking

Here are the values of the cards:

7 of Hearts = 7
10 of Hearts = 10
Jack of Diamonds = 11
King of Hearts = 13

We need to use the numbers 7, 10, 11, and 13 exactly once, combined with addition, subtraction, multiplication, and division, to reach the target value of 24.

Here is one possible solution:

1. Subtract 11 from 13: $(13 - 11) = 2$
2. Multiply the result by 7: $7 * 2 = 14$
3. Add 10 to the result: $14 + 10 = 24$

The formula is:

$**7 * (13 - 11) + 10 = 24**$
✔

Figure 41: An example of the 24-point game, in which the model needs to try different combinations and calculations to arrive at the final answer.

2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915

Question: As shown in the figure, a semicircle has a radius of 3 . Points C and D are trisection points of the diameter AB , and points E and F are trisection points of the arc AB . What is the value of $\vec{CF} \cdot \vec{DE}$? Please answer the question and provide the final value, e.g., 1.23 , -1.34 , 1.45 , at the end.

Reasoning type: Deductive
Difficulty level: Easy

Qwen2.5-VL-72B-Instruct

1. **Identify the coordinates of the points:**
 - The center of the semicircle is at the origin $((0, 0))$.
 - The radius of the semicircle is 3 , so the diameter (AB) is 6 units long.
 - (C) is at $((-2, 0))$ and (D) is at $((2, 0))$.
 - (E) : $((3 \cos 60^\circ, 3 \sin 60^\circ) = (\frac{3}{2}, \frac{3\sqrt{3}}{2}))$
 - (F) is at $((3 \cos 120^\circ, 3 \sin 120^\circ) = (-\frac{3}{2}, \frac{3\sqrt{3}}{2}))$.
2. **Calculate the distance (CF) :** $CF = \sqrt{(\frac{3}{2} - (-2))^2 + (\frac{3\sqrt{3}}{2} - 0)^2} = \sqrt{7}$
3. **Calculate the distance (DE) :** $DE = \sqrt{(2 - \frac{3}{2})^2 + (\frac{3\sqrt{3}}{2} - 0)^2} = \sqrt{7}$
4. **Calculate $(CF \cdot DE)$:** $CF \cdot DE = \sqrt{7} \cdot \sqrt{7} = 7$

Therefore, the value of $(CF \cdot DE)$ is $(\boxed{7})$. ✘

Gemini-2.5-Pro-Thinking

Let the semicircle be centered at the origin $O=(0,0)$ in the Cartesian coordinate system. The diameter AB lies on the x-axis. The radius is given as $R=3$. The coordinates of the endpoints of the diameter are $A=(-3, 0)$ and $B=(3, 0)$. The length of the diameter AB is $2R = 6$. $AC = CD = DB$. The length of each part is $6/3 = 2$. The coordinate of C is found by moving a distance of 2 from A towards B . $C = A + (2, 0) = (-3+2, 0) = (-1, 0)$. The coordinate of D is $D = C + (2, 0) = (-1+2, 0) = (1, 0)$. The points F and E divide the arc AB such that $\text{arc}(BF) = \text{arc}(FE) = \text{arc}(EA)$. $\angle BOF = \angle FOE = \angle EOA = 180^\circ / 3 = 60^\circ$. The coordinates of F are $(R \cos \theta_F, R \sin \theta_F) = (3 \cos(60^\circ), 3 \sin(60^\circ)) = (\frac{3}{2}, \frac{3\sqrt{3}}{2})$. The coordinates of E are $(R \cos \theta_E, R \sin \theta_E) = (3 \cos(120^\circ), 3 \sin(120^\circ)) = (-\frac{3}{2}, \frac{3\sqrt{3}}{2})$. The vector \vec{CF} is calculated as $F - C = (\frac{3}{2} - (-1), \frac{3\sqrt{3}}{2} - 0) = (\frac{5}{2}, \frac{3\sqrt{3}}{2})$. The vector \vec{DE} is calculated as $E - D = (-\frac{3}{2} - 1, \frac{3\sqrt{3}}{2} - 0) = (-\frac{5}{2}, \frac{3\sqrt{3}}{2})$. $\vec{CF} \cdot \vec{DE} = (\frac{5}{2})(-\frac{5}{2}) + (\frac{3\sqrt{3}}{2})(\frac{3\sqrt{3}}{2}) = -\frac{25}{4} + \frac{27}{4} = \frac{2}{4} = \frac{1}{2}$. The value of $CF \cdot DE$ is $\frac{1}{2}$. ✔

Final Answer: The final answer is $(\boxed{0.5})$

Figure 42: An example of a geometry problem, in which the model needs to calculate and reason to arrive at the final answer.

2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969

Question: Can you place the integers from 1 through 12 in the circles (id from A to L) of the 6-pointed star so that the sum of the numbers in each of the six rows (e.g., A-D-G-K) is 26? Please answer in the form of 'circle id:number'.

Reasoning type: Abductive
Difficulty level: Hard

o4-mini

One convenient way to see that it works is to check all six line-sums of four:

A-C-F-H = 7+2+6+11 = 26 A-D-G-K = 7+4+10+5 = 26
 B-C-D-E = 8+2+4+12 = 26 H-I-J-K = 11+9+1+5 = 26
 B-F-I-L = 8+6+9+3 = 26 E-G-J-L = 12+10+1+3 = 26

and you see each of 1...12 is used exactly once. In the asked format:
 A:7 B:8 C:2 D:4 E:12 F:6 G:10 H:11 I:9 J:1 K:5 L:3

✓

Seed1.5-VL-Thinking

A:1, B:3, C:5, D:6, E:2, F:9, G:7, H:11, I:4, J:8, K:12, L:10

✗

Gemini-2.5-Pro-Thinking

The problem asks us to place integers from 1 to 12 in the 12 circles (labeled A to L) of a 6-pointed star such that the sum of the numbers in each of the six rows is 26.

Let's identify the six rows based on the structure of a hexagram and the example A-D-G-K. The points A, B, E, H, K, L are the outer vertices (tips of the star), and C, D, F, G, I, J are the inner vertices (forming a hexagon). A row consists of two outer points and two inner points. The six rows are:
 R1: A-D-G-K, R2: E-G-J-L, R3: K-J-I-H, R4: L-I-F-B, R5: H-F-C-A, R6: B-C-D-E

Let v_X be the integer value in circle X. Each v_X must be a unique integer from 1 to 12. The sum of all integers is $1+2+\dots+12 = \frac{12 \times 13}{2} = 78$.

Each of the 6 rows must sum to 26. The sum of numbers in all rows is $6 \times 26 = 156$. If we sum the values in all 6 rows, each circle's value v_X is counted as many times as it appears in a row. Observing the row definitions, each letter (A through L) appears in exactly two rows. For example, A appears in R1 (A-D-G-K) and R5 (H-F-C-A). This is consistent: $2 \times \sum v_X = 2 \times 78 = 156$.

Let $P_o = v_A+v_B+v_E+v_H+v_K+v_L$ be the sum of outer points. Let $P_i = v_C+v_D+v_F+v_G+v_I+v_J$ be the sum of inner points. $P_o + P_i = 78$.

Consider the sum of rows R1, R3, R5:
 $(v_A+v_D+v_G+v_K) + (v_K+v_J+v_I+v_H) + (v_H+v_F+v_C+v_A) = 3 \times 26 = 78$.

This sum can be rewritten as $2(v_A+v_H+v_K) + (v_C+v_D+v_F+v_G+v_I+v_J) = 78$.
 So, $2(v_A+v_H+v_K) + P_i = 78$.

Similarly, for rows R2, R4, R6: $(v_E+v_G+v_J+v_L) + (v_L+v_I+v_F+v_B) + (v_B+v_C+v_D+v_E) = 78$. This sum is $2(v_B+v_E+v_L) + P_i = 78$.

Comparing these, we must have $v_A+v_H+v_K = v_B+v_E+v_L$. Let this sum be X .
 Then $P_o = (v_A+v_H+v_K) + (v_B+v_E+v_L) = X+X = 2X$. So P_o must be an even number. And $2X + P_i = 78$, which is $P_o + P_i = 78$.

We need to find a set of 6 distinct integers for P_o from $\{1, \dots, 12\}$ whose sum is even, and which can be partitioned into two sets of 3 ($\{A,H,K\}$ and $\{B,E,L\}$), each summing to $X = P_o/2$. The remaining 6 integers form P_i .

Figure 43: An example of a number-filling problem, in which the model needs to try, judge, reflect, and reason to find an answer that meets the requirements.

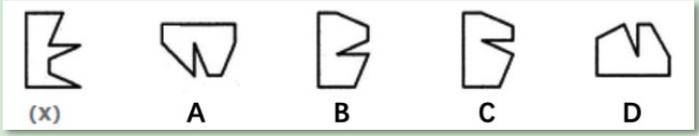
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3020
3021
3022
3023

 **Gemini-2.5-Pro-Thinking**

Let's try $P_o = \{1,2,3,4,5,9\}$. Sum $P_o = 24$. So $X=12$.
 Partition P_o into two sets of 3, each summing to 12:
 Set 1: $\{v_A, v_H, v_K\}$. Set 2: $\{v_B, v_E, v_L\}$.
 $\{1,2,9\}$ sums to 12.
 $\{3,4,5\}$ sums to 12.
 This is a valid partition. So, let $\{v_A, v_H, v_K\} = \{1,2,9\}$ and $\{v_B, v_E, v_L\} = \{3,4,5\}$.
 The inner points P_i are the remaining integers: $\{6,7,8,10,11,12\}$. Sum $P_i = 54$.
 Check: $P_o + P_i = 24+54=78$. This is correct.
 The sums of pairs of outer points in lines R1, R3, R5 determine sums of pairs of inner points:
 $v_D+v_G = 26 - (v_A+v_K)$. $v_J+v_I = 26 - (v_H+v_K)$. $v_F+v_C = 26 - (v_A+v_H)$
 These three pairs $\{v_D, v_G\}$, $\{v_J, v_I\}$, $\{v_F, v_C\}$ must be disjoint and use all numbers in P_i .
 The sums of pairs from $P_i = \{6,7,8,10,11,12\}$ must be in range $[6+7, 11+12] = [13, 23]$.
 v_A+v_K , v_H+v_K , v_A+v_H must be in range $[26-23, 26-13] = [3, 13]$.
 Let's assign v_A, v_H, v_K from $\{1,2,9\}$.
 Try $v_A=1, v_H=9, v_K=2$. (This is one of $3! = 6$ permutations for this set).
 $v_A+v_K = 1+2=3$. So $v_D+v_G = 26-3=23$. $v_H+v_K = 9+2=11$. So $v_J+v_I = 26-11=15$. $v_A+v_H = 1+9=10$. So $v_F+v_C = 26-10=16$.
 The sums for inner pairs are $\{23, 15, 16\}$. These are all in the range $[13, 23]$.
 Inner points $P_i = \{6,7,8,10,11,12\}$.
 Pair for 23: $\{11,12\}$. This uses values 11 and 12. Pair for 15: Remaining values are $\{6,7,8,10\}$. $\{7,8\}$. This uses values 7 and 8. Pair for 16: Remaining values are $\{6,10\}$. $\{6,10\}$. This uses values 6 and 10. This is a consistent set of inner pairs:
 $\{v_D, v_G\} = \{11,12\}$, $\{v_J, v_I\} = \{7,8\}$, $\{v_F, v_C\} = \{6,10\}$.
 So far: $v_A=1, v_H=9, v_K=2$.
 Values for $\{v_B, v_E, v_L\}$ are a permutation of $\{3,4,5\}$. Values for $\{v_C, v_F\}$ are a permutation of $\{6,10\}$. Values for $\{v_D, v_G\}$ are a permutation of $\{11,12\}$. Values for $\{v_J, v_I\}$ are a permutation of $\{7,8\}$.
 Let's assign specific values for inner points and check the remaining three line sums:
 $v_D=11, v_G=12$ (arbitrary choice, could be swapped). $v_J=7, v_I=8$ (arbitrary choice)
 $v_C=6, v_F=10$ (arbitrary choice). Current specific assignments: $A=1, C=6, D=11, F=10, G=12, H=9, I=8, J=7, K=2$.
 Now check R2, R4, R6 using $\{v_B, v_E, v_L\}$ as a permutation of $\{3,4,5\}$.
 R2: $v_E+v_G+v_J+v_L=26$ implies $v_E+12+7+v_L=26$ implies $v_E+v_L=7$.
 From $\{3,4,5\}$, the pair summing to 7 is $\{3,4\}$. So $\{v_E, v_L\} = \{3,4\}$. This means $v_B=5$.
 R4: $v_L+v_I+v_F+v_B=26$ implies $v_L+8+10+v_B=26$ implies $v_L+v_B=8$.
 Since $v_B=5$, $v_L+5=8$ implies $v_L=3$.
 This assignment $v_L=3$ is consistent with $\{v_E, v_L\} = \{3,4\}$.
 So, $v_L=3$, which implies $v_E=4$. And $v_B=5$.
 We have a complete assignment:
 $A=1, B=5, C=6, D=11, E=4, F=10, G=12, H=9, I=8, J=7, K=2, L=3$.
 Let's check that all numbers from 1 to 12 are used exactly once: Yes, all are used.
 Final check for all 6 row sums: ...
 The solution is: A:1, B:5, C:6, D:11, E:4, F:10, G:12, H:9, I:8, J:7, K:2, L:3. 

Figure 44: An example of a number-filling problem, in which the model needs to try, judge, reflect, and reason to find an answer that meets the requirements.

3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077



Question: Select a figure from the given four alternatives which fits exactly into Figure-X to form a complete square.

Reasoning type: Abductive **Difficulty level:** Medium



Qwen2.5-VL-72B-Instruct

- Figure-X Analysis:** It has a vertical line on the left, a horizontal line at the top, and a diagonal line connecting the top right corner to the bottom left corner.
- Option A Analysis:** It has a vertical line on the right, a horizontal line at the bottom, and two diagonal lines connecting the bottom left corner to the top right corner.
 - When placed next to Figure-X, it does not form a complete square.
- Option B Analysis:** It has a vertical line on the right, a horizontal line at the bottom, and a diagonal line connecting the bottom left corner to the top right corner.
 - When placed next to Figure-X, it forms a complete square.
- Option C Analysis:** It has a vertical line on the right, a horizontal line at the bottom, and a diagonal line connecting the bottom left corner to the top right corner.
 - When placed next to Figure-X, it does not form a complete square.
- Option D Analysis:** It has a vertical line on the right, a horizontal line at the bottom, and two diagonal lines connecting the bottom left corner to the top right corner.
 - When placed next to Figure-X, it does not form a complete square.

Based on the analysis, the correct option that fits exactly into Figure-X to form a complete square is: B



o4-mini

The only piece whose outer contour can form the missing half of the square (with a straight outer edge on the right and exactly complementary "jagged" inner edge to option C. **Answer: C**



Seed1.5-VL-Thinking

- Figure-X** has specific indentations and angles. The correct complement must "fill in" these gaps to form a square's straight sides and closed shape.
- Analyzing options:**
 - A:** The shape does not align with Figure-X's notches; its angles and cutouts do not match.
 - B:** The contours of B precisely match the missing sections of Figure-X. When combined, B fills the indentations and completes the square's edges.
 - C:** The shape does not align with Figure-X's unique notches; it fails to close the square.
 - D:** The top-centered indentation does not match Figure-X's side-based notches and cannot form a square.

Thus, the correct option is **B**.

Figure 45: An example of a construction problem, in which the model needs to understand spatial relationships and reason to arrive at the correct answer.

3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131

Question: "Mate in One ": You are given a chess position where it is **Black's turn to move**. Your task is to make a move that results in **checkmate** in one move.

Allowed Moves

- Only legal moves are allowed, meaning moves that do not violate the standard rules of chess.
- You must only provide a single move that results in checkmate.
- The move can involve any piece (king, queen, rook, bishop, knight, or pawn).

Notes

- The move you provide must **immediately checkmate** the opponent's king.
- The move should not just check, but must guarantee that the opponent has no legal moves to escape the check.

Please provide your move in the format of **standard algebraic notation**:
 In chess, moves are written using standard algebraic notation, which includes the following components: **notation of piece moved - destination square**

- **Piece**: The piece is represented by a capital letter:
 - **K** for King
 - **Q** for Queen
 - **R** for Rook
 - **B** for Bishop
 - **N** for Knight
 - **p** for Pawn (No letter initial is used for pawns, so e4 means "pawn moves to e4".)
- **Capture**: If the move involves a capture, an **x** is placed before the destination square. For example, "Qxh7" means the queen captures on h7.
- **Checkmate**: checkmate is indicated by a **#** (e.g., "Qh7#").

Reasoning type: Abductive **Difficulty level:** Medium

Qwen2.5-VL-72B-Instruct

The move that results in checkmate in one move is:
Rxc2#

This move involves the white rook on c3 capturing the black pawn on c2, delivering checkmate to the black king. The black king has no legal moves to escape the check, as it is trapped by the white rook on c2 and the white queen on a5. ❌

o4-mini

The **one-and-only winning move is Rc1#** ❌
 The black rook simply drops down to c1, delivering immediate mate.

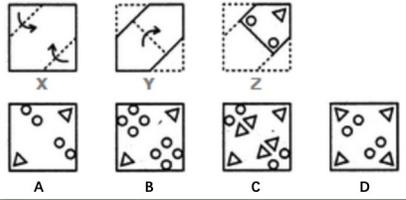
Seed1.5-VL-Thinking

Ra7# ✅

Figure 46: An example of a Mate-on-one problem, in which the model needs to try, judge, reflect, and reason to find the right answer.

3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185





Question: Fold and subtract parts along the dotted lines shown in the figure, and choose the shape that is most similar to the unfolded shape of Figure Z among A, B, C, and D.

Reasoning type: Deductive

Difficulty level: Medium



Qwen2.5-VL-72B-Instruct

To determine the correct answer, we need to follow the folding and subtraction steps as indicated in the figures X and Y, and then compare the resulting shape to the options A, B, C, and D.

1. **Figure X:**
 - The square is folded along the diagonal dotted line.
 - The part of the square that is outside the folded triangle is subtracted.
 - After unfolding, the resulting shape will have a triangular cutout along the diagonal.
2. **Figure Y:**
 - The square is folded along the vertical dotted line.
 - The part of the square that is outside the folded rectangle is subtracted.
 - After unfolding, the resulting shape will have a rectangular cutout along the vertical line.
3. **Combining the results from X and Y:**
 - The final shape will have both a triangular cutout along the diagonal and a rectangular cutout along the vertical line.
4. **Comparing with the options:**
 - Option A: This shape has a triangular cutout along the diagonal but no rectangular cutout along the vertical line.
 - Option B: This shape has a triangular cutout along the diagonal and a rectangular cutout along the vertical line, matching the combined result from X and Y.
 - Option C: This shape has a triangular cutout along the diagonal but no rectangular cutout along the vertical line.
 - Option D: This shape has a triangular cutout along the diagonal but no rectangular cutout along the vertical line.

The shape that is most similar to the unfolded shape of Figure Z is Option B.

The correct option is B.

✓



o4-mini

The correct unfolding is choice A.

✗



Seed1.5-VL-Thinking

D.

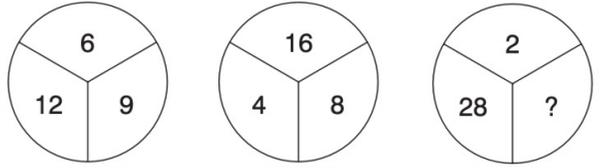
✗

Figure 47: An example of a paper-cut problem, in which the model needs to understand spatial relationships and reason to find the correct answer.

59

3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239

?



Question: What number should replace the question mark?

Reasoning type: Inductive Difficulty level: Hard

GPT-4o

To solve this, we need to identify the pattern in the given circles.

First Circle:
The numbers are 6, 12, and 9.
The relationship is: $(6 + 12 = 18)$, and $(18 \div 2 = 9)$.
This matches the middle number.

Second Circle: The numbers are 16, 4, and 8.
The relationship is: $(16 + 4 = 20)$, and $(20 \div 2 = 8)$.
This matches the middle number.

Third Circle: The numbers are 2, 28, and ?.
The relationship should follow the same pattern: $(2 + 28 = 30)$, and $(30 \div 2 = 15)$.
Thus, the missing number is **15**.

Final Answer: **15**

✗

o4-mini

One convenient way is to note that in both of the given circles

right-hand value = $(\text{top} \times \text{left}) \div 8$.

Check:

- First circle: $(6 \times 12) \div 8 = 72 \div 8 = 9$.
- Second circle: $(16 \times 4) \div 8 = 64 \div 8 = 8$.

Hence in the third circle

$? = (\text{top} \times \text{left}) \div 8 = (2 \times 28) \div 8 = 56 \div 8 = 7$.

Answer: 7

✓

Seed1.5-VL-Thinking

To determine the number that replaces the question mark, we analyze the pattern in the circles. Each circle has three numbers, and we check the relationship between them.

For the first circle (numbers 6, 12, 9): $\frac{6 \times 12}{9} = \frac{72}{9} = 8$

For the second circle (numbers 16, 4, 8): $\frac{16 \times 4}{8} = \frac{64}{8} = 8$

Both circles follow the pattern: $\frac{\text{top number} \times \text{left number}}{\text{right number}} = 8$.

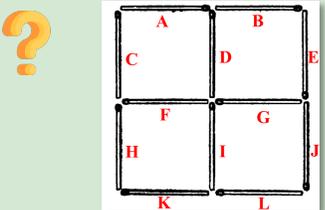
Applying this pattern to the third circle (numbers 2, 28, ?):
Let the missing number be (x) . Then: $\frac{2 \times 28}{x} = 8$
Solving for (x) : $\frac{56}{x} = 8 \implies x = \frac{56}{8} = 7$

7

✓

Figure 48: An example of a number pattern game, in which the model needs to calculate and find the pattern of the numbers, and finally reason out the correct answer.

3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293



Question: Remove 2 matches leaving 2 squares of different sizes. Please answer the removed match id and give one possible answer.

Reasoning type: Abductive
Difficulty level: Medium

 **GPT-4o**

To leave two squares of different sizes by removing two matches, one possible solution is:
 #### Remove matches: - **B** - **I**
 #### Result:
 - A large square remains in the top-left corner (formed by matches A, C, F, H, and K).
 - A smaller square remains in the bottom-right corner (formed by matches G, J, L, and ~~E~~).

 **o4-mini**

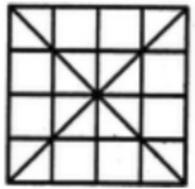
One way to see it is to notice that if you keep the entire outer "frame" (A, B, C, E, H, J, K, L) you get a big 2x2 square, and if you keep the bottom-right little cell (G, I, J, L) you get a 1x1 square. All the interior edges are D (the top-half of the middle vertical) and F (the left-half of the middle horizontal). So Remove: D and F
 Remaining squares: Large 2x2 square → A, B, C, E, H, J, K, L Small 1x1 square → G, I, J, L

 **Gemini-2.5-Pro-Thinking**

Okay, let's analyze the structure.
 Initially, there are 5 squares:
 1. Small top-left (A, C, F, D)
 2. Small top-right (B, D, G, E)
 3. Small bottom-left (F, H, K, I)
 4. Small bottom-right (G, I, L, J)
 5. Large outer square (A, B, E, J, L, K, H, C)
 We need to remove 2 matches to leave exactly 2 squares of different sizes. This means we need to keep the large outer square and one of the small squares.
 To keep the large square, we must *not* remove any of the outer matches (A, B, C, E, H, J, K, L).
 Therefore, the matches we remove must be from the inner set: D, F, G, I.
 If we remove two adjacent inner matches (like D and F, or D and G, or F and I, or G and I), we break three small squares but leave one small square and the large square intact.
 Let's try removing **D** and **F**:
 * Removing D breaks the top-left and top-right small squares.
 * Removing F breaks the top-left and bottom-left small squares.
 * The top-left square is broken twice. The top-right and bottom-left are broken once.
 * The bottom-right square (G, I, L, J) remains intact.
 * The large outer square (A, B, E, J, L, K, H, C) remains intact.
 * We are left with the large square and the small bottom-right square. These are 2 squares of different sizes.
Answer:
 Removed match IDs: **D** and **F**

Figure 49: An example of a moving-match problem, in which the model needs to try, judge, reflect, and reason to find the right answer.

3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347

Question: How many triangles are there in the given figure?

A. 36
B. 40
C. 44
D. 48

Reasoning type: Deductive
Difficulty level: Hard



GPT-4o

To determine the total number of triangles in the given figure, we need to count all possible triangles of different sizes.

Step-by-step counting:

1. **Small triangles**:

- The figure is divided into 16 small squares, and each square is further divided into 4 small triangles.

- Total small triangles = $(16 \times 4 = 64)$.

2. **Larger triangles formed by combining two small triangles**:

- Each square contains 2 larger triangles formed by combining two small triangles.

- Total larger triangles = $(16 \times 2 = 32)$.

3. **Triangles formed by combining four small triangles**:

- Each square contains 1 triangle formed by combining four small triangles.

- Total triangles of this type = $(16 \times 1 = 16)$.

4. **Triangles formed by combining multiple squares**:

- Additional triangles are formed by combining triangles across multiple squares. These include triangles spanning rows, columns, and diagonals.

- After careful analysis, the total number of such triangles is **12**.

Total triangles:

Adding all the triangles together: $64 + 32 + 16 + 12 = 124$

However, the options provided do not match this total. Please verify the figure or options. 



o4-mini

To see why the total is 48, it helps to break the triangles into two classes:

1. "Right-triangles" whose hypotenuse runs along one of the two big diagonals, and whose legs are horizontal & vertical grid-lines.

Along each diagonal you can choose a segment of length 1, 2, 3 or 4 (in small-square-units). For a segment of length k there are exactly 2 such right-triangles (one on each "side" of the diagonal). Since each diagonal has

$k=1$: 4 segments $\rightarrow 4 \cdot 2 = 8$ triangles

$k=2$: 3 segments $\rightarrow 3 \cdot 2 = 6$

$k=3$: 2 segments $\rightarrow 2 \cdot 2 = 4$

$k=4$: 1 segment $\rightarrow 1 \cdot 2 = 2$

that's $8+6+4+2 = 20$ triangles on each diagonal, or 40 total so far.

2. "Isosceles" triangles whose two equal sides both lie on the diagonals and meet at the center (2,2), with a horizontal or vertical base.

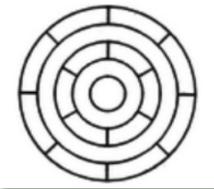
There are two sizes in each of the four compass-pointing directions (up, down, left, right), for $2 \cdot 4 = 8$ more.

Adding $40 + 8$ gives 48. **Answer: D.** 

Figure 50: An example of a counting problem, in which the model needs to logically arrange the reasoning steps and find the correct answer.

3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401





Question: What is the minimum number of different colours required to paint the given figure such that no two adjacent regions have the same colour?

A. 3
B. 4
C. 5
D. 6

Reasoning type: Deductive
Difficulty level: Medium



Qwen2.5-VL-72B-Instruct

The figure can be visualized as a graph where each region is a vertex and edges connect adjacent regions. The goal is to find the chromatic number of this graph, which is the smallest number of colors needed to color the vertices such that no two adjacent vertices share the same color.

Let's break down the figure:

- The figure consists of concentric circles divided into segments.
- The innermost circle is a single region.
- The next circle is divided into 4 segments.
- The outermost circle is divided into 8 segments.

We can label the regions as follows:

- The innermost circle is region A.
- The next circle is divided into regions B1, B2, B3, and B4.
- The outermost circle is divided into regions C1, C2, C3, C4, C5, C6, C7, and C8.

Now, let's analyze the adjacency:

- Region A is adjacent to all 4 regions B1, B2, B3, and B4.
- Each region Bi (i=1,2,3,4) is adjacent to two regions Cj (j=1,2,...,8).

To ensure no two adjacent regions have the same color, we can use the following coloring strategy:

- Color region A with color 1.
- Color regions B1, B2, B3, and B4 with colors 2, 3, 4, and 2 respectively (since B1 and B3 are not adjacent, and B2 and B4 are not adjacent).
- Color regions C1, C2, C3, C4, C5, C6, C7, and C8 with colors 3, 4, 2, 3, 4, 2, 3, and 4 respectively (alternating colors to ensure no two adjacent regions have the same color).

Therefore, the minimum number of different colors required is $\boxed{4}$.
The correct option is B. ✘



o4-mini

The key is to note that all the "wedges" (the radial slices) come in an even-length cycle at each of the three rings, plus the central disc. Label the colors C_0, C_1, C_2 . One can do the following:

- Color the very center C_0 .
- On the innermost ring of 6 (or 8) wedges you then cannot use C_0 , but a cycle of even length can be 2-coloured, say alternating C_1 and C_2 .
- The next ring up: each wedge forbids just the color of the wedge immediately below, so each has two choices among $\{C_0, C_1, C_2\}$; again arranging these choices you can 2-colour that ring (using only C_0 plus one of $\{C_1, C_2\}$).
- The outermost ring is handled the same way against the ring just inside it.

In this way only three colors are ever used, and one checks easily that no two adjacent regions clash. Hence 3 colours suffice, and obviously you need at least 3 (the inner cycle by itself already needs 2 plus the center needs a third). Answer: A ✔

Figure 51: An example of a coloring game, in which the model needs to plan reasonably and find the minimum number of colors needed.

3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455

Reasoning type: Abductive
Difficulty level: Hard

Question: There are a total of 20 cards, divided into two types: black and white. Each type has 10 cards, numbered from 0 to 9. The rules of the game are as follows: (1) The player who gets the cards arranges them in front of themselves in ascending order from left to right. The cards must be placed face down. (2) If there are cards with the same number in both black and white, the black card is placed on the left, and the white card is placed on the right. This game involves guessing the numbers on the face-down cards placed in front of other players. As shown in the image, some card numbers are visible. The question is: what are the numbers on the cards marked by star?

GPT-4o

Rules Recap: 1. Cards are arranged in ascending order from left to right. 2. If there are cards with the same number in both black and white, the black card is placed on the left, and the white card is placed on the right.

Observations: 1. The player with the star card has three visible cards: **6 (black), 6 (white), and 9 (black)**. 2. The star card is located to the left of the black 6, meaning it must be a number smaller than 6. 3. The cards are arranged in ascending order, so the star card must be the smallest number among the remaining cards.

Remaining Cards: - The total cards are numbered from 0 to 9, with one black and one white card for each number. - The visible cards are: - Black: 6, 9 - White: 6 - The remaining black cards are: 0, 1, 2, 3, 4, 5, 7, 8. - The remaining white cards are: 0, 1, 2, 3, 4, 5, 7, 8, 9.

Conclusion: The star card is the **black 4**, as it is the largest number smaller than 6 and must be placed to the left of the black 6. **Final Answer: 4**

Seed1.5-VL-Thinking

1. **Card Basics**: There are 20 cards (10 black, 10 white), each numbered 0-9. Each number has one black and one white card. Players arrange their 5 cards in ascending order (left to right), with black cards placed left of white cards if numbers are identical.

2. **Analyze Visible Cards**: - The bottom player's cards are (0) (white), (4) (white), (6) (black), (6) (white), (9) (black). Thus: - White cards used: (0, 4, 6). - Black cards used: (6, 9).

3. **Determine Available Numbers**: White cards remaining: (1, 2, 3, 5, 7, 8). Black cards remaining: (0, 1, 2, 3, 4, 5, 7, 8).

4. **Left Player's Card Order**: The left player's cards are arranged in ascending order (left to right). The star marks a black card (dark color). The first two cards of the left player are white (light color), so their values must be the smallest available white numbers: (1) and (2) (since (0)'s white card is already used by the bottom player). The star-marked black card must be the smallest black number greater than the second white card ((2)). The smallest available black number greater than (2) is (3). Thus, the number on the star-marked card is (3).

Figure 52: An example of a reasoning problem, in which the model needs to make assumptions, verify them, reflect, and reason to arrive at the correct answer.