

REFGRADER: AUTOMATED GRADING OF MATHEMATICAL COMPETITION PROOFS USING AGENTIC WORKFLOWS

**Hamed Mahdavi¹ Pouria Mahdavinia¹ Samira Malek¹ Pegah Mohammadipour¹
Alireza Hashemi² Majid Daliri³ Alireza Farhadi⁴ Amir Khasahmadi⁵
Niloofar Mireshghallah⁶ Vasant Honavar¹**

¹Pennsylvania State University ²City University of New York ³New York University
⁴Amirkabir University of Technology ⁵Autodesk ⁶Carnegie Mellon University

ABSTRACT

State-of-the-art (SOTA) LLMs have progressed from struggling on proof-based Olympiad problems to solving most of the IMO 2025 problems, with leading systems reportedly handling 5 of 6 problems. Given this progress, we assess how well these models can grade proofs: detecting errors, judging their severity, and assigning fair scores beyond binary correctness. We study proof-analysis capabilities using a corpus of 90 Gemini 2.5 Pro-generated solutions that we grade on a 1–4 scale with detailed error annotations, and on MathArena solution sets for IMO/USAMO 2025 scored on a 0–7 scale. Our analysis shows that models can reliably flag incorrect (including subtly incorrect) solutions but exhibit calibration gaps in how partial credit is assigned. To address this, we introduce agentic workflows that extract and analyze reference solutions and automatically derive problem-specific rubrics for a multi-step grading process. We instantiate and compare different design choices for the grading workflows, and evaluate their trade-offs. Across our annotated corpus and MathArena, our proposed workflows achieve higher agreement with human grades and more consistent handling of partial credit across metrics. We release all code, data, and prompts/logs to facilitate future research. <https://github.com/ref-grader/ref-grader>

1 INTRODUCTION

Until early 2025, state-of-the-art (SOTA) LLMs often failed to produce correct and sound solutions to Olympiad level problems (Petrov et al., 2025; Mahdavi et al., 2025). Industry announcements from Google and OpenAI claimed that the advanced versions of their models could achieve gold medal level performance on the IMO 2025, solving 5 of 6 problems within exam time (Luong & Lockhart, 2025; Wei). Independent reproductions report solving 5 of 6 problems using Gemini 2.5 Pro within an agentic, multi-step workflow (Huang & Yang, 2025). As automated judges, they performed unreliably, often near chance, when asked to distinguish invalid solutions from the correct ones or to apply rubrics consistently (Mahdavi et al., 2025; Petrov et al., 2025).

These findings raise concerns about using LLMs for automated proof assessment: if models struggle with basic verification and rubric application, automatic grading may be unreliable. However, the cited studies predate recent model advances. Independent evaluations, such as Balunović et al. (2025), report notable improvements in solution correctness and proof quality generated by SOTA (non-agentic) LLMs (e.g., Gemini 2.5 Pro), hinting at their potential improvement in proof verification. Evaluating LLMs’ mathematical capabilities via final-answer accuracy has become the de facto standard (Cobbe et al., 2021; Hendrycks et al., 2021; Fang et al., 2024; Yue et al., 2024). Going beyond final answers to assess proof quality is substantially more challenging. Formal verification offers a principled solution to validation (Zheng et al., 2022; Lin et al., 2025; Chen et al., 2025; Jiang et al., 2024; Ren et al., 2025), but faces two practical limitations: limited availability of formal corpora and lower readability for humans. An alternative is to binarize proofs and measure agreement with expert judges (Dekoninck et al., 2025; Guo et al., 2025), which improves scalability but ignores

the issue of partial credits. We emphasize that **partial credit assignment** will be an **increasingly important capability** as we move towards more complex LLM-based proof generation systems.

We constructed a corpus of 90 carefully selected problems from **complex IMO shortlist problems**, alongside Gemini 2.5 Pro-generated solutions for each problem, graded on a 1–4 scale and annotated with precise error types and locations to act as **rich ground truth for partial credit assignment**. We also use the data gathered from the MathArena IMO/USAMO 2025 solutions scored 0–7. Using Gemini 2.5 Pro with maximum thinking budget, we first assess single-turn grading by comparing model-assigned scores against human grades. Next, we introduce **Agentic Workflows** that extract and analyze reference solutions to **automatically design problem-specific grading rubrics (Ref-Grader)**, and we compare design choices: approachability-based weighting (by “aha moment” difficulty), milestone-based rubrics, their hybrid, and a 3-step reference variant without rubric induction. **Our workflows substantially improve upon single-turn grading** in partial-credit grading across diverse metrics such as Pearson/Spearman, MAE/RMSE, QWK and AC2. **We validate robustness through systematic ablations and cross-dataset evaluation**. While our workflows may require greater token consumption (both input and output) and thus incur higher costs, the majority of the workflow steps are cacheable, which helps keep the overall cost low.

2 RELATED WORK

Proof-evaluation corpora: Benchmarks assessing proofs include the Open Proof Corpus, which aggregates human and model proofs with binary validity labels and expert annotations (Dekoninck et al., 2025), and LitmusTest, which standardizes pass/fail judgments using expert-designed rubrics (Guo et al., 2025). For competition mathematics, MathArena hosts model-generated solutions for IMO/USAMO-style problems with 0-7 scores and judge rationales (Balunović et al., 2025). Formal settings emphasize verifiable correctness but face constraints in data availability and coverage (Lin et al., 2025; Zheng et al., 2022; Chen et al., 2025).

LLM-as-a-grader: Two directions are prominent: rubric-grounded grading across domains and reliability improvements via calibration or multi-agent designs. Recent work spans diverse applications. In physics education, GPT-4o assigns partial credit using self-consistency and human-in-the-loop triage (Chen & Wan, 2025). In healthcare, open-ended clinical dialogues are evaluated against physician-written, instance-specific criteria (Arora et al., 2025). For expert long-form tasks, expert-validated rubrics map to checklist items (Ruan et al., 2025), while rubric-prompted judge distributions benefit from calibration to human ratings (Hashemi et al., 2024). In education and code assessment, rubric specialization and multi-agent judging improve robustness and interpretability (Pathak et al., 2025; Chu et al., 2025). Per-problem rubrics have been used to diagnose stepwise skills on word problems (Jin et al., 2024).

LLM-as-a-judge: Complementary work examines models as evaluators to reduce dependence on human annotations (Stephan et al., 2024; Li et al., 2024; Nasrabadi, 2024; Ning et al., 2024). Recent methods treat assessment as adaptable and task-aware (Tan et al., 2024; Dhurandhar et al., 2024) and calibrate reliability against human judgments (Kim et al., 2024; Ye et al., 2024; Liu et al., 2025). General-purpose evaluation resources include UltraFeedback (Cui et al., 2024), AlpacaEval (Dubois et al., 2024), Chatbot Arena (Chiang et al., 2024), and MT-Bench (Zheng et al., 2023). For mathematics specifically, judge benchmarks include REASONEVAL (Xia et al., 2025), MATHCHECK (Zhou et al., 2024), and SMART-840 (Cherian et al., 2024).

Reasoning Benchmarks: A range of datasets evaluate mathematical reasoning in large language models (LLMs) (Ahn et al., 2024), spanning arithmetic-only benchmarks (Yuan et al., 2023) and math word problem (MWP) datasets like GSM8K (Cobbe et al., 2021) and MathQA (Amini et al., 2019) that require logical reasoning (Wei et al., 2022); related robustness and compositionality benchmarks include GSM1K, Compositional GSM, and Functional MATH (Zhang et al., 2024; Hosseini et al., 2024; Srivastava et al., 2024), while automated theorem proving (ATP) datasets target formal theorem proving (Zheng et al., 2022; Yu et al., 2024; Jiang et al., 2024); advanced and Olympiad-level evaluations include CONIC10K, GHOSTS, miniGHOSTS, CHAMP, OlympiadBench, MathOdyssey, and Omni-MATH (Wu et al., 2023; Frieder et al., 2023; Mao et al., 2024; He et al., 2024; Fang et al., 2024; Gao et al., 2024), and complementary resources include HARP and NuminaMath (Yue et al., 2024; LI et al., 2024).

Mathematical Reasoning in LLMs: Reasoning can be elicited through prompting and inference-time strategies, including Chain-of-Thought and self-consistency (Chen et al., 2024; Wei et al., 2022; Kojima et al., 2023; Havrilla et al., 2024; Wang et al., 2023; Wang & Zhou, 2024). Controlled benchmarks reveal gaps between pattern matching and formal reasoning (Hendrycks et al., 2021; Mirzadeh et al., 2024). Complementary work explores reward modeling, self-refinement, and algorithmic decomposition (Huang et al., 2024; Zelikman et al., 2023).

3 DATASETS

3.1 IMO SHORTLIST DATA

Data Collection. We selected 90 challenging problems from the IMO Shortlist dataset (2017-2023). We used a standardized prompt requesting a rigorous solution to each Olympiad-level problem and generated one solution per problem with Gemini 2.5 Pro. The prompt is provided in Appendix A. We then annotated the solutions using the fallacy categories from (Mahdavi et al., 2025). The list of fallacies is as follows:

- **Proof by Example**
- **Proposal Without Verification**
- **Inventing Wrong Facts**
- **Begging the Question (Circular Reasoning)**
- **Solution by Trial-and-Error**
- **Calculation Mistakes**

We adopt the definitions provided in the original paper (Mahdavi et al., 2025). We additionally introduce a general category, **Wrong Logical Conclusion**, to tag mathematical errors that do not fit any of the other categories. Evaluators carefully reviewed each solution and annotated each error type and the approximate error location using the following syntax (markup used in the released dataset):

```
<span class="[Fallacy Type]+"> [Fallacious Statement] </span>
```

For example, if a fallacy is identified in a generated proof, evaluators mark it as follows:

```
<span class="proof-by-example"> As the statement is true for  $n = 1, 2, 3$  it is highly probable that it is also true </span>
```

When applying fallacy labels, if multiple fallacies fit a given error, we prioritized the most relevant one. When distinct errors co-occurred, we applied multiple fallacy labels. We graded solutions using the following 4-point scale.

- **1: Incorrect:** The solution does not contain useful non-trivial information. It contains only incorrect information or restates straightforward facts from the problem. Equivalent to 0/7 or 1/7 in Olympiad grading.
- **2: Some Correct Information:** The solution contains a few non-trivial facts derived with some effort but lacks a coherent proof. Equivalent to 2/7 or 3/7 in Olympiad grading.
- **3: Almost Correct:** The solution proves non-trivial parts of the argument but omits one non-trivial part of the proof. Equivalent to 4/7 or 5/7 in Olympiad grading.
- **4: Correct:** The solution proves all required facts and statements

We did not adopt the 0-7 Olympiad scale due to the higher probability of inconsistencies between human evaluators, and considerably more human effort to grade at that granularity. Finally, after annotating errors and assigning grades, evaluators provided a brief explanation of any issues in a dataset field labeled "Final Comment".

Dataset Statistics. Figures 1a, 1b and 1c summarize dataset statistics: error frequencies by fallacy category, the distribution of solution labels, and the topical composition of problems. Relative to the models analyzed by Mahdavi et al. (2025), Gemini 2.5 Pro yields a smaller share of incorrect solutions (Fig. 1b) and fewer naive errors (e.g., Proof by Example, Solution by Trial-and-Error; Fig. 1a).

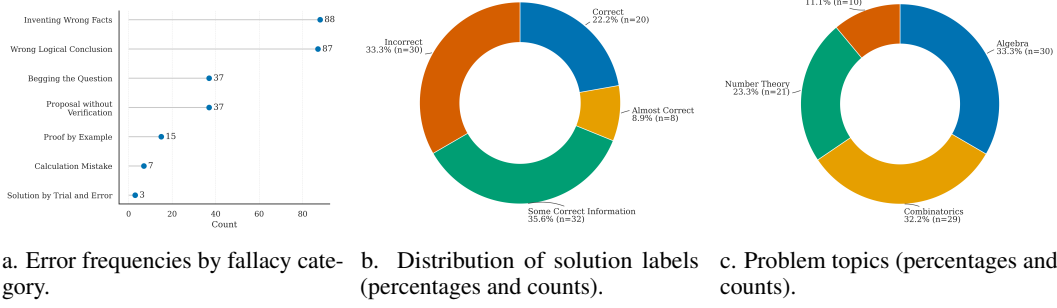


Figure 1: Dataset summaries and error analysis for the IMO Shortlist dataset

3.2 MATHARENA DATA

We collected 385 solutions for IMO and USAMO 2025 from the MathArena website. The solutions were generated by the following models: Grok 3 (Think), DeepSeek-R1-0528, Gemini 2.5 Pro, Gemini 2.0 Flash Thinking, QwQ-32B, DeepSeek-R1, o1-pro (high), o3-mini (high), o4-mini (high), Grok 4, o3 (high), and Claude-3.7-Sonnet (Think). MathArena conducts independent evaluations of model performance on contest-level problems. Solutions are graded by human judges on a 0-7 scale. The MathArena grade distribution is zero-inflated because many model-generated solutions receive a zero on these challenging problems. To balance the dataset for analysis and visualization, we subsampled zero-scores with probability 0.14 (applying this subsample consistently in the figures and tables for this section). Figure 2 shows the resulting grade distribution.

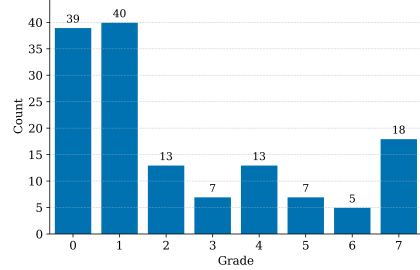


Figure 2: Grade distribution for the MathArena dataset

4 EVALUATION SETTING

Our goal is to evaluate LLMs as graders of mathematical proofs on the IMO Shortlist and MathArena datasets. Let $\mathcal{D} = \{(p_i, s_i)\}_{i=1}^n$ denote problem-solution pairs with associated ground-truth grades $\{g_i\}_{i=1}^n$. For each instance i , let $R_i = \{r_{ij}\}_{j=1}^{m_i}$ denote the set of correct reference solutions. The grading procedure (agentic workflow) takes (p_i, s_i, R_i) as input and outputs a predicted grade $\hat{g}_i$. For all experiments, the end result is an LLM output in a structured format that includes the predicted grade $\hat{g}_i$ and, when available, step-by-step analysis, identified errors, clarity/structure/notation tags, and constructive feedback.

To assess agreement between $\{\hat{g}_i\}$ and $\{g_i\}$, we report Pearson and Spearman correlations, mean absolute error (MAE), root mean squared error (RMSE), off-by-one and off-by-two tolerance rates, quadratic weighted kappa (QWK), and Gwet's AC2.

The first four metrics are well-known and we omit their definitions here. We define the off-by-one and off-by-two metrics as:

$$\text{Off-by-one} = \frac{1}{n} \sum_{i=1}^n \mathbf{1}\{|g_i - \hat{g}_i| \leq 1\}, \quad \text{Off-by-two} = \frac{1}{n} \sum_{i=1}^n \mathbf{1}\{|g_i - \hat{g}_i| \leq 2\}.$$

These summarize near-miss accuracy when small deviations are acceptable but large errors are costly.

Quadratic weighted kappa (QWK). QWK (Cohen, 1968) measures agreement on ordinal labels while accounting for chance. With K grade categories, let $O, E \in \mathbb{R}^{K \times K}$ be the observed and

expected confusion matrices, and let $w_{ij} = (i - j)^2 / (K - 1)^2$. Then

$$\kappa = 1 - \frac{\sum_{i,j} w_{ij} O_{ij}}{\sum_{i,j} w_{ij} E_{ij}}.$$

Under rater independence, the expected matrix E uses the raters' marginal grade distributions p and q , with entries $E_{ij} = p_i q_j$ (or $n p_i q_j$ for counts), where $p_i = \sum_j O_{ij}$ and $q_j = \sum_i O_{ij}$. This sets the chance baseline. When marginals are skewed, the baseline is high and QWK can be low even if raw agreement is high.

Gwet's AC2. AC2 (Gwet, 2014) uses the same ordinal weights but a chance model that is less sensitive to skew. It replaces the independence baseline $p_i q_j$ with a pooled marginal distribution π_i computed across raters (for two raters with marginals p and q , $\pi_i = (p_i + q_i)/2$, or $(n_i^A + n_i^B)/(2n)$ with counts), and computes expected weighted disagreement $D_e = \sum_{i,j} w_{ij} \pi_i \pi_j$. With observed weighted disagreement $D_o = \sum_{i,j} w_{ij} P_{ij}$ for the normalized table P ,

$$\text{AC2} = 1 - \frac{D_o}{D_e}.$$

Using pooled marginals reduces the impact of imbalanced category frequencies, making AC2 typically more stable than QWK when categories are skewed.

Because grades are discrete and skewed, **no single metric is sufficiently reliable on its own**, therefore, we draw conclusions from a comprehensive analysis across all metrics. Pearson and Spearman track association but not error cost, and both can look strong when a model predicts the most frequent grade. Ties on a coarse 0 to 7 scale further reduce Spearman's resolution. MAE is interpretable in points and RMSE highlights large mistakes, yet either can look good if the model mostly predicts the same common grade. Off-by-one and off-by-two quantify near-miss tolerance but can be inflated by always predicting central or majority grades. QWK is ordinal and chance-corrected, but it is sensitive to skewed marginals and can show a paradox: raw agreement can be high yet QWK low when grade frequencies are highly imbalanced, because the chance baseline is also high. AC2 keeps ordinal weights but uses pooled marginals, making it more robust to imbalance. It is typically more stable than QWK when categories are skewed, though results still depend on the weight choice and very rare grades. Since no metric is perfect, we report them together to get a balanced view of ordering, error size, near-miss tolerance, and agreement beyond chance in this imbalanced setting.

For the IMO Shortlist, we map the 4-point scale to the 0-7 scale using $m(x) = 2x - 1$ for $x \in \{1, 2, 3, 4\}$. MathArena is already on the 0-7 scale.

5 EXPERIMENTAL RESULTS

We first evaluate the performance of LLMs for single-turn proof grading and present quantitative metrics alongside qualitative visualizations.

5.1 SINGLE-TURN GRADING

In our first experiment, we focus on evaluating the performance of LLMs on grading proofs in a single-turn setting. We add the problem and solution in the context and ask the LLM to analyze the proof step-by-step, find all of its errors, and then grade the proof on a 0-7 scale. We use the following definition for the grading scale:

Definition	Score	Definition	Score
No progress	0	Substantial progress	4
Understaning trace	1	One small flaw	5
Minor progress	2	Negligible issues	6
Partial progress	3	Perfect	7

The full grading prompt used in this setting is provided in Appendix B. The results for MathArena and the IMO Shortlist dataset are shown in Table 1. For reference, the **standard deviation** and

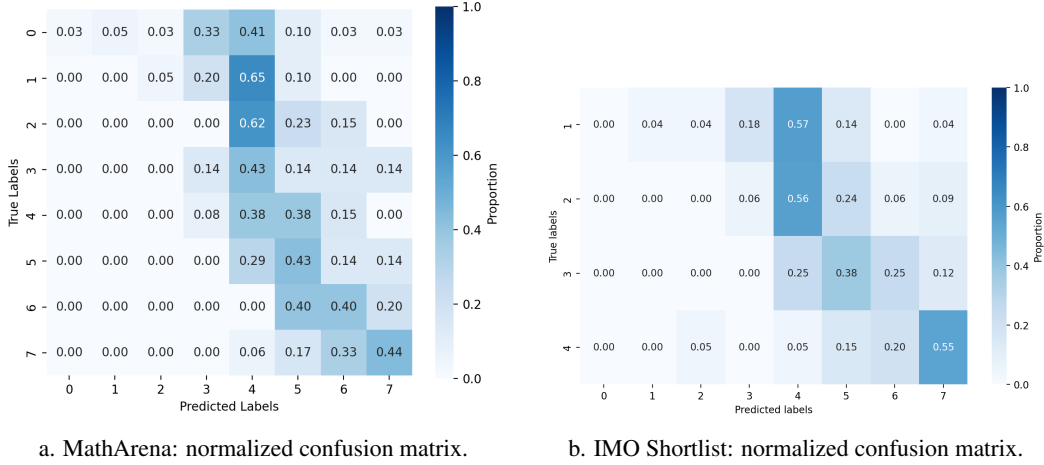


Figure 3: Normalized confusion matrices for single-turn grading on MathArena and IMO Shortlist.

mean absolute deviation of real scores are **2.42** and **1.87** for MathArena, and **2.22** and **1.68** for the IMO Shortlist. If we always predict the majority grade, **Off-by-one** and **Off-by-two** are 0.56 and 0.65 on MathArena, and 0.31 and 0.68 on the IMO Shortlist, respectively. Hence, although the correlation metrics indicate non-random association, single-turn grader struggles to predict the real scores accurately.

Figures 3a and 3b show normalized confusion matrices. On both datasets, the grader tends to over-score very low-grade solutions and partially correct ones (grades 0-4), shifting probability mass to the right of the diagonal. By contrast, solutions with grades ≥ 5 show a stronger diagonal. This pattern is consistent with the findings of Dekoninck et al. (2025) and Guo et al. (2025). Under a binarized evaluation (correct vs not correct), performance would be high. More specifically, most off-diagonal mass concentrates a few bins to the right of the true grade for true grades 0-3, indicating an optimistic bias and a tendency to credit incomplete outlines. Misclassifications are predominantly adjacent, and there are fewer solutions with grades 2-6 in both datasets due to data imbalance, which yields higher rank-based measures (Pearson/Spearman) while increasing absolute error (MAE/RMSE). In the bottom-right region, over-scoring is limited, yielding a clearer diagonal and explaining the strong binary separation at threshold 5.

Conceptually, binary grading is simpler: a reliable verifier can confirm the correctness of a complete solution and identify shortcomings in incomplete or incorrect solutions. However, assessing progress in incomplete solutions is more challenging. Assigning fair partial credit is ambiguous when the model cannot solve the problem using the solution’s attempted approach, even if it can solve the problem by a different approach, that alone is not sufficient. We show empirically that leveraging **a set of candidate reference solutions** within a multi-step grading workflow yields substantially better performance.

Dataset	Pearson $\uparrow$	Spearman $\uparrow$	MAE $\downarrow$	RMSE $\downarrow$	QWK $\uparrow$	Off1 $\uparrow$	Off2 $\uparrow$	AC2 $\uparrow$
Math-Arena	0.665	0.633	2.324	2.745	0.359	0.317	0.486	0.357
IMO Shortlist	0.601	0.596	1.756	2.211	0.427	0.500	0.689	0.479

Table 1: Single-turn grading results. Higher is better for correlations, QWK, Off1/Off2/AC2. Lower is better for MAE/RMSE.

5.2 MULTI-STEP GRADING WITH REFERENCE SOLUTIONS

We next evaluate reference-aided, multi-step grading workflows and ablations. To address the conceptual issue discussed above, we introduce a multi-step reference grading workflow (*Ref-Grader*). We collected a large set of reference solutions for both the IMO Shortlist and MathArena datasets

from the [AoPS](#) forum. We use the following workflow that exploits reference solutions to improve the quality and calibration of grading:

1. **Reference Solution Clustering:** The model clusters the reference solutions into groups based on their similarity.
2. **Solution Matching:** The model finds the most similar group of reference solutions to the given solution and uses it as a reference to grade the given solution.
3. **Solution Analysis:** The model analyzes the reference solution and breaks it into the main steps based on the "aha moments (main ideas of the solution)" and its substeps.
4. **Rubric Design:** The model distributes 7 points among the main steps and defines rules on how to allocate points to the substeps.
5. **Grading:** The model detects errors in two ways: (1) direct error detection, or (2) contradictions with the reference solution. Contradictions imply the given solution is wrong at that step. Then the model matches the correct and errorneous parts of the given solution with the rubrics and decides the final grade.

The schema of the workflow is shown in Figure 4. Each of the steps above is a single model call with a specific prompt. Prompts for all steps are provided in Appendix B.1.

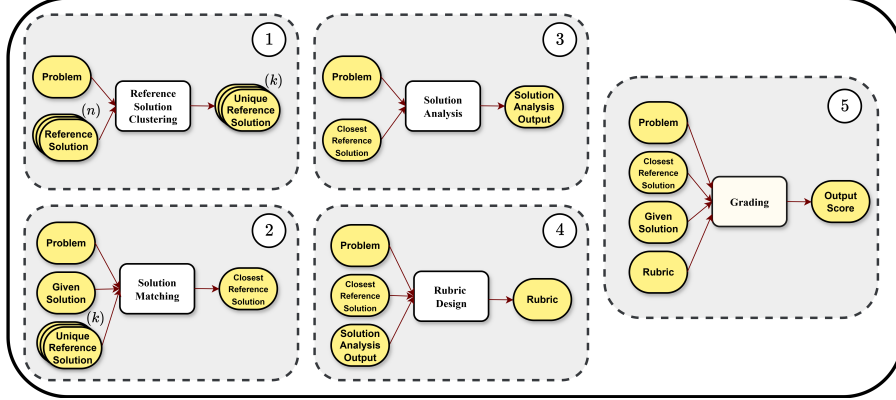


Figure 4: The high-level schema of our multi-stage grading workflow

Ablations and settings. To study the role of each component, we compare the *Single-turn Grader* (one model call without reference solutions), the *5-step Ref-Grader (Plain)* (full workflow with reference solutions, solution analysis, and rubric design), the *5-step Ref-Grader (Approachability)*, which in step 3 computes step-level approachability scores (1-5, measuring how hard a main step is to be chosen) and in step 4 allocates rubric points proportional these scores, the *5-step Ref-Grader (Milestones)*, which in step 4 designs the rubric by milestones reached (milestones denote proving the same or an equivalent intermediate statement as in the reference solution up to a specific step) and the *5-step Ref-Grader (Hybrid)*, which combines the approachability-based analysis with the milestone-based rubric.

We also evaluate the *3-step Ref-Grader (No Rubrics)*, in which step 3 uses a single-turn grading prompt with the reference solution added, without solution analysis and rubric design (Figure 5 illustrates this variant). This ablation isolates the effect of adding a reference solution and highlights the additional contributions of solution analysis and rubric design.

Tables 2 and 3 summarize the results. Across both datasets, the 5-step Ref-Grader (Approachability) and 5-step Ref-Grader (Milestones) achieve the strongest metrics overall. The 5-step Ref-Grader (Plain) typically ranks third. The 3-step Ref-Grader (No Rubrics) outperforms the Single-turn Grader

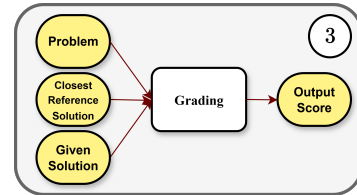


Figure 5: Workflow: reference solution clustering, solution matching, and grading.

on most metrics, indicating that adding a similar reference solution helps even without explicit rubric generation. Overall, these results show that both the reference solution and the rubric contribute to grader performance significantly, and that more careful rubric design brings additional gains. The pearson and rank correlation metrics are shift invariant, as a result, they don't account for the cases where we get the associations right while we are systematically underestimating or overestimating the scores. MAE, RMSE, Off-by-1 and Off-by-2 denote the gaps between real and predicted scores. Finally, QWK and AC2 consider both association and the gap between real and predicted scores.

Interestingly, the 5-step Ref-Grader (Hybrid) performs worse than the other 5-step variants, likely because approachability interferes with milestones: approachability is a property of a reference solution's step and it assigns a score based on the approach of the reference solution, whereas a milestone can be independent of a specific reference solution, so the two notions are not fully compatible. As a practical note, steps 1 (reference clustering), 3 (solution analysis), and 4 (rubric design) **can be cached offline**, as they do not depend on the specific given solution. Thus, only steps 2 and 5 need to run online per each given solution. This amortizes the cost of the 5-step workflow.

Method	r ↑	ρ ↑	MAE ↓	RMSE ↓	QWK ↑	Off1 (↑)	Off2 (↑)	AC2 (↑)
Single-turn Grader	0.66	0.63	2.32	2.75	0.36	0.32	0.49	0.36
3-step Ref-Grader (No Rubrics)	0.71	0.72	2.27	2.66	0.42	0.29	0.51	0.37
5-step Ref-Grader (Plain)	0.73	0.75	1.49	2.09	0.67	0.63	0.77	0.63
5-step Ref-Grader (Approachability)	0.79	0.77	1.33	1.98	0.72	0.67	0.81	0.68
5-step Ref-Grader (Milestones)	0.78	0.71	1.26	1.89	0.73	0.63	0.81	0.72
5-step Ref-Grader (Hybrid)	0.76	0.76	1.50	2.12	0.68	0.61	0.74	0.63

Table 2: MathArena: Single-turn vs multi-step reference grading.

Method	r ↑	ρ ↑	MAE ↓	RMSE ↓	QWK ↑	Off1 (↑)	Off2 (↑)	AC2 (↑)
Single-turn Grader	0.60	0.60	1.76	2.21	0.43	0.50	0.69	0.48
3-step Ref-Grader (No Rubrics)	0.70	0.71	1.58	2.04	0.55	0.51	0.78	0.57
5-step Ref-Grader (Plain)	0.73	0.74	1.26	1.83	0.70	0.66	0.83	0.75
5-step Ref-Grader (Approachability)	0.73	0.74	1.19	1.75	0.72	0.69	0.83	0.77
5-step Ref-Grader (Milestones)	0.73	0.72	1.20	1.80	0.71	0.68	0.86	0.77
5-step Ref-Grader (Hybrid)	0.66	0.65	1.36	1.97	0.64	0.63	0.80	0.71

Table 3: IMO Shortlist: Single-turn vs multi-step reference grading.

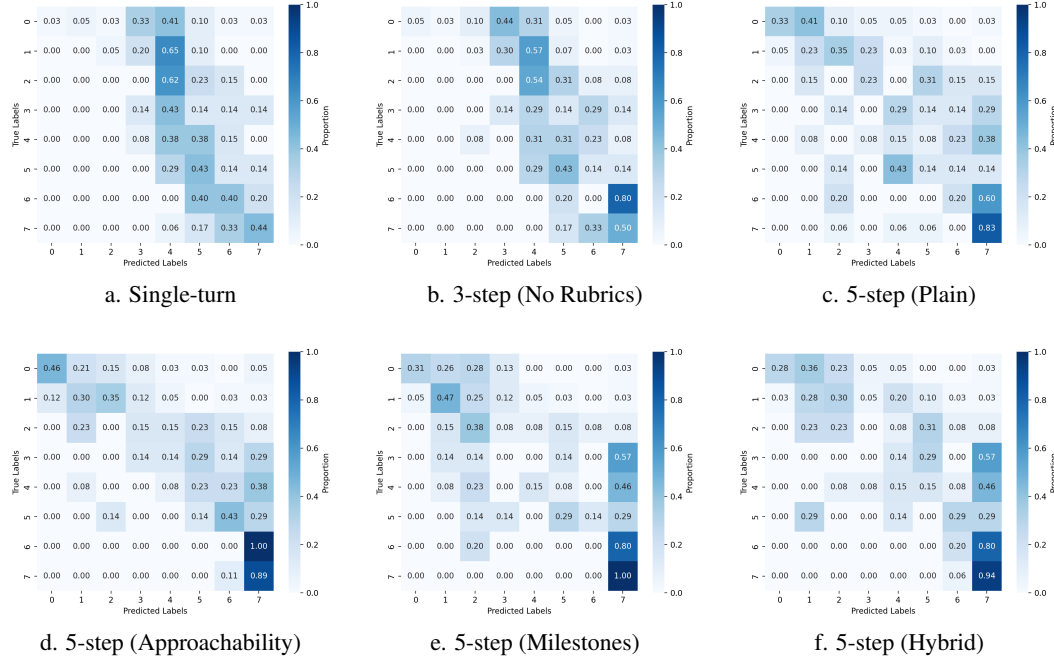
5.3 ARE WE DETECTING PROGRESS?

Our main promise is to detect partial progress in incomplete solutions. To make this concrete, we examine the normalized confusion matrices for each method on MathArena and the IMO Shortlist (Figure 6). In an ideal scenario, probability mass should be concentrated along the diagonal. We define partial-progress solutions as those with true score above 0 for MathArena and score label above 1 for the IMO Shortlist. As seen for the single-turn grader earlier, it struggles to produce calibrated scores. By looking at the predicted scores for the solutions with partial progress, several patterns emerge. The single-turn grader almost never produces scores below 3, and using a hard threshold on its output does not reliably separate zero-progress from non-zero-progress solutions. The 3-step RefGrader (No Rubrics) alleviates this to some extent: probability mass shifts toward lower scores for zero-progress solutions and toward higher predicted scores for higher true scores, though calibration issues remain visible in the confusion matrix.

The 5-stage workflows look different, and a diagonal pattern starts to appear, albeit imperfectly. For these workflows, probability mass for low-score solutions shifts toward their true scores, and solutions with true scores of 0 and 1 for MathArena and 1 for the IMO Shortlist can be discriminated with high accuracy by applying a threshold to the predicted output scores.

At the other extreme, one can also ask whether complete solutions can be separated from incomplete ones. For MathArena, all models distinguish the 6 and 7 solutions from the rest quite well. In the context of math olympiads, a 6 solution is a complete solution that contains a minor issue, so discriminating between a 6 and a 7 solution can depend on taste and rubric design rather than an

MathArena



IMO Shortlist

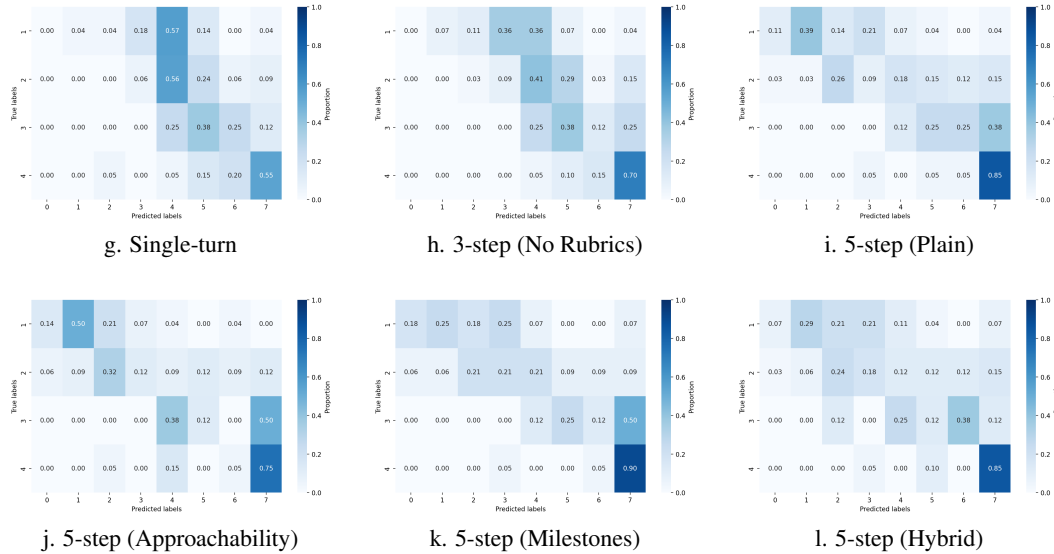


Figure 6: Normalized confusion matrices by each method for MathArena and IMO Shortlist.

objective mathematical criterion. The same pattern is seen for the IMO Shortlist. All methods assign scores 6 and 7 to solutions with the 4 (correct) label. This further supports our earlier point about the difficulty of separating solutions with progress from those without in comparison to the task of discriminating perfect solutions from incomplete and imperfect ones.

This analysis provides a consistent story for the observed metrics in the previous subsection. The single-turn grader is able to detect perfectly correct solutions versus imperfect ones, but it struggles to detect partial progress. Since all solutions in both datasets are LLM-generated, the solutions are

skewed toward zero-progress and completeness; hence, the single-turn grader achieves significant association because it can detect perfectly complete solutions versus imperfect ones. Adding the reference solution and the rubric to the workflow adds the capability of detecting partial progress, which further improves associational and non-associational metrics.

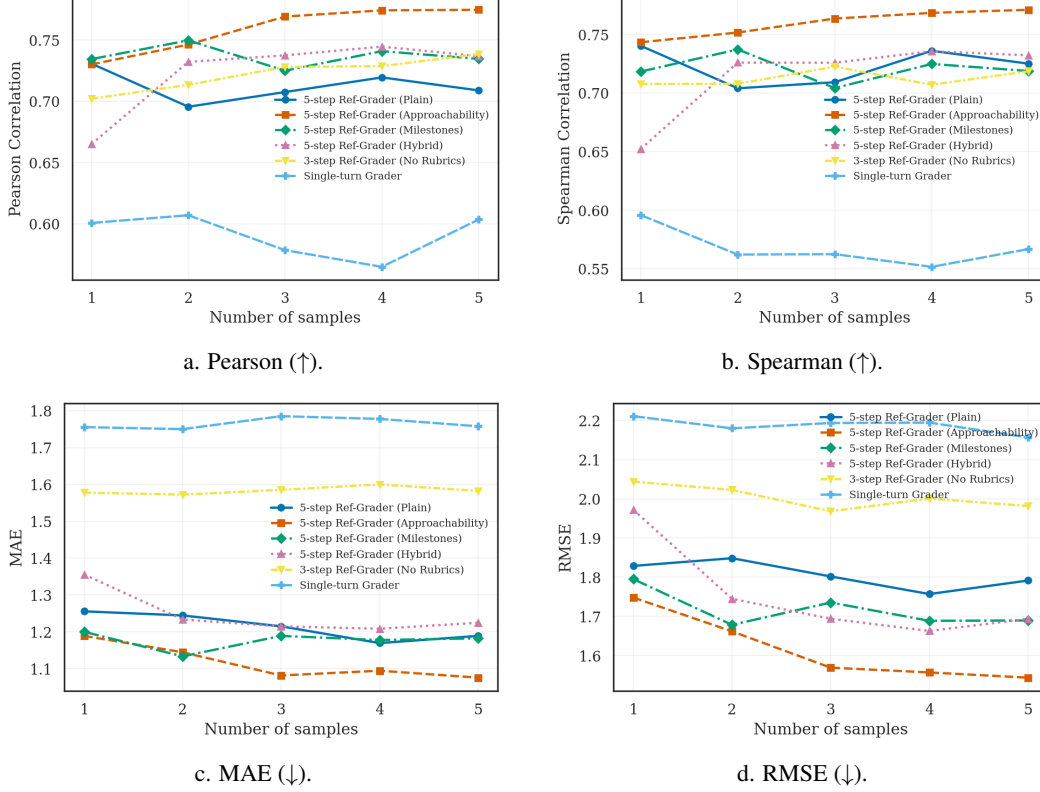


Figure 7: Sampling trends for the grader steps across methods for the IMO Shortlist dataset. As we can see, sampling and averaging the grader steps does not consistently add much benefit except for the 5-step Ref-Grader (Approachability).

5.4 SAMPLING AND AVERAGING

We mentioned that the multi-step grading workflow costs more than the single-turn grading workflow. It is natural to ask whether sampling and averaging the outputs of the single-turn grader explains the gains. Figure 7 plots sampling trends for all workflows. With the exception of the 5-step Ref-Grader (Approachability), within-method sampling and averaging yields no consistent performance gains, indicating that improvements are not due to spending more tokens relative to the single-turn grader.

Interestingly, ensembling across methods can also help. For the IMO Shortlist, averaging predictions from 3-step Ref-Grader (No Rubrics), 5-step Ref-Grader (Approachability), 5-step Ref-Grader (Plain), and 5-step Ref-Grader (Milestones) achieves Pearson 0.80, Spearman 0.80, MAE 1.11, RMSE 1.52, off-by-one 0.65, and off-by-two 0.82, matching or exceeding the best single-method metrics. A systematic study of ensembling strategies is left for future work.

6 CONCLUSION

We studied proof grading for Olympiad-level mathematics and showed that reference-aided, multi-step workflows substantially improve partial-credit calibration over single-turn graders. Across the IMO Shortlist and MathArena datasets, our 5-step Ref-Grader variants consistently increase agreement with human judges, with approachability-weighted and milestone-based rubrics offering

complementary strengths. Ablations indicate that adding a similar reference solution helps even without rubric induction, while sampling/averaging within a method does not explain the gains.

Beyond evaluation, these workflows support broader uses. First, as LLM-as-a-judge, they provide transparent, step-referenced rationales and more stable partial-credit decisions than rubric-free judging. Second, as a generative reward model for reinforcement learning, the rubric-informed, reference-grounded scoring can shape trajectories toward correct and complete proofs. Third, in education, the same approach can grade student work and surface interpretable feedback on missing steps and error types, provided appropriate reference solutions and guardrails are available. We release data, code, and prompts to facilitate adoption and extensions.

REFERENCES

- Janice Ahn, Rishu Verma, Renze Lou, Di Liu, Rui Zhang, and Wenpeng Yin. Large language models for mathematical reasoning: Progresses and challenges, 2024. URL <https://arxiv.org/abs/2402.00157>.
- Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. Mathqa: Towards interpretable math word problem solving with operation-based formalisms, 2019. URL <https://arxiv.org/abs/1905.13319>.
- Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. Healthbench: Evaluating large language models towards improved human health, 2025. URL <https://arxiv.org/abs/2505.08775>.
- Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions, February 2025. URL <https://matharena.ai/>.
- Haolin Chen, Yihao Feng, Zuxin Liu, Weiran Yao, Akshara Prabhakar, Shelby Heinecke, Ricky Ho, Phil Mui, Silvio Savarese, Caiming Xiong, and Huan Wang. Language models are hidden reasoners: Unlocking latent reasoning capabilities via self-rewarding, 2024. URL <https://arxiv.org/abs/2411.04282>.
- Luoxin Chen, Jinming Gu, Liankai Huang, Wenhao Huang, Zhicheng Jiang, Allan Jie, Xiaoran Jin, Xing Jin, Chenggang Li, Kaijing Ma, Cheng Ren, Jiawei Shen, Wenlei Shi, Tong Sun, He Sun, Jiahui Wang, Siran Wang, Zhihong Wang, Chenrui Wei, Shufa Wei, Yonghui Wu, Yuchen Wu, Yihang Xia, Huajian Xin, Fan Yang, Huaiyuan Ying, Hongyi Yuan, Zheng Yuan, Tianyang Zhan, Chi Zhang, Yue Zhang, Ge Zhang, Tianyun Zhao, Jianqiu Zhao, Yichi Zhou, and Thomas Hanwen Zhu. Seed-prover: Deep and broad reasoning for automated theorem proving, 2025. URL <https://arxiv.org/abs/2507.23726>.
- Zhongzhou Chen and Tong Wan. Grading explanations of problem-solving process and generating feedback using large language models at human-level accuracy. *Phys. Rev. Phys. Educ. Res.*, 21: 010126, Mar 2025. doi: 10.1103/PhysRevPhysEducRes.21.010126. URL <https://doi.org/10.1103/PhysRevPhysEducRes.21.010126>.
- Anoop Cherian, Kuan-Chuan Peng, Suhas Lohit, Joanna Matthiesen, Kevin Smith, and Joshua B. Tenenbaum. Evaluating large vision-and-language models on children’s mathematical olympiads, 2024. URL <https://arxiv.org/abs/2406.15736>.
- Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Hao Zhang, Banghua Zhu, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating llms by human preference, 2024. URL <https://arxiv.org/abs/2403.04132>.
- Yucheng Chu, Hang Li, Kaiqi Yang, Harry Shomer, Yasemin Copur-Gencturk, Leonora Kaldaras, Kevin Haudek, Joseph Krajcik, Namsu Shin, Hui Liu, and Jiliang Tang. A llm-powered automatic grading framework with human-level guidelines optimization. In Caitlin Mills, Giora Alexandron, Davide Taibi, Giosuè Lo Bosco, and Luc Paquette (eds.), *Proceedings of the 18th International Conference on Educational Data Mining (EDM 2025)*, pp. 31–41, Palermo, Italy,

- July 2025. International Educational Data Mining Society. ISBN 978-1-7336736-6-2. doi: 10.5281/zenodo.15870201. URL <https://educationaldatamining.org/EDM2025/proceedings/2025.EDM.long-papers.80/index.html>.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.
- Jacob Cohen. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. *Psychological Bulletin*, 70(4):213–220, 1968. doi: 10.1037/h0026256.
- Ganqu Cui, Lifan Yuan, Ning Ding, Guanming Yao, Bingxiang He, Wei Zhu, Yuan Ni, Guotong Xie, Ruobing Xie, Yankai Lin, Zhiyuan Liu, and Maosong Sun. Ultrafeedback: Boosting language models with scaled ai feedback, 2024. URL <https://arxiv.org/abs/2310.01377>.
- Jasper Dekoninck, Ivo Petrov, Kristian Minchev, Mislav Balunovic, Martin Vechev, Miroslav Marinov, Maria Drencheva, Lyuba Konova, Milen Shumanov, Kaloyan Tsvetkov, Nikolay Drenchev, Lazar Todorov, Kalina Nikolova, Nikolay Georgiev, Vanesa Kalinkova, and Margulan Ismoldayev. The open proof corpus: A large-scale study of llm-generated mathematical proofs, 2025. URL <https://arxiv.org/abs/2506.21621>.
- Amit Dhurandhar, Rahul Nair, Moninder Singh, Elizabeth Daly, and Karthikeyan Natesan Ramamurthy. Ranking large language models without ground truth, 2024. URL <https://arxiv.org/abs/2402.14860>.
- Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled alpacaEval: A simple way to debias automatic evaluators, 2024. URL <https://arxiv.org/abs/2404.04475>.
- Meng Fang, Xiangpeng Wan, Fei Lu, Fei Xing, and Kai Zou. Mathodyssey: Benchmarking mathematical problem-solving skills in large language models using odyssey math data, 2024. URL <https://arxiv.org/abs/2406.18321>.
- Simon Frieder, Luca Pinchetti, Alexis Chevalier, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Christian Petersen, and Julius Berner. Mathematical capabilities of chatgpt, 2023. URL <https://arxiv.org/abs/2301.13867>.
- Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, et al. Omni-math: A universal olympiad level mathematic benchmark for large language models. In *International Conference on Learning Representations (ICLR) — OpenReview*, 2024. URL <https://openreview.net/forum?id=yaqPf0KAlN>.
- Dadi Guo, Jiayu Liu, Zhiyuan Fan, Zhitao He, Haoran Li, Yumeng Wang, and Yi R. Fung. Mathematical proof as a litmus test: Revealing failure modes of advanced large reasoning models, 2025. URL <https://arxiv.org/abs/2506.17114>.
- Kilem L. Gwet. *Handbook of Inter-Rater Reliability: The Definitive Guide to Measuring the Extent of Agreement Among Raters*. Advanced Analytics, LLC, Gaithersburg, MD, 4 edition, 2014. ISBN 9780970806284.
- Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13806–13834, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.745. URL <https://aclanthology.org/2024.acl-long.745/>.
- Alex Havrilla, Sharath Raparthy, Christoforus Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskiy, Eric Hambro, and Roberta Raileanu. GloRE: When, where, and how to improve llm reasoning via global and local refinements, 2024. URL <https://arxiv.org/abs/2402.10963>.

Chaoqun He, Renjie Luo, Yuzhuo Bai, Shengding Hu, Zhen Leng Thai, Junhao Shen, Jinyi Hu, Xu Han, Yujie Huang, Yuxiang Zhang, Jie Liu, Lei Qi, Zhiyuan Liu, and Maosong Sun. Olympiad-bench: A challenging benchmark for promoting agi with olympiad-level bilingual multimodal scientific problems, 2024.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset, 2021. URL <https://arxiv.org/abs/2103.03874>.

Arian Hosseini, Alessandro Sordoni, Daniel Kenji Toyama, Aaron Courville, and Rishabh Agarwal. Not all llm reasoners are created equal. In *Proceedings of the 4th Workshop on Mathematical Reasoning and AI (MATH-AI) at NeurIPS 2024*, 2024. URL <https://openreview.net/forum?id=RcqAmkDJfI>. Introduces the Compositional GSM benchmark.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. Large language models cannot self-correct reasoning yet, 2024. URL <https://arxiv.org/abs/2310.01798>.

Yichen Huang and Lin F. Yang. Gemini 2.5 pro capable of winning gold at imo 2025, 2025. URL <https://arxiv.org/abs/2507.15855>.

Dongwei Jiang, Marcio Fonseca, and Shay B. Cohen. Leanreasoner: Boosting complex logical reasoning with lean, 2024. URL <https://arxiv.org/abs/2403.13312>.

Hyoungwook Jin, Yoonsu Kim, Yeon Su Park, Bekzat Tilekbay, Jinho Son, and Juho Kim. Using large language models to diagnose math problem-solving skills at scale. In *L@S 2024 - Proceedings of the 11th ACM Conference on Learning @ Scale*, L@S 2024 - Proceedings of the 11th ACM Conference on Learning @ Scale, pp. 471–475. Association for Computing Machinery, Inc, July 2024. doi: 10.1145/3657604.3664697.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Cristina Grau-Vilchez, Yik Siu Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Cynthia Breazeal, and Hae Won Park. A demonstration of adaptive collaboration of large language models for medical decision-making, 2024. URL <https://arxiv.org/abs/2411.00248>.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2023. URL <https://arxiv.org/abs/2205.11916>.

Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: A comprehensive survey on llm-based evaluation methods, 2024. URL <https://arxiv.org/abs/2412.05579>.

Jia LI, Edward Beeching, Lewis Tunstall, Ben Lipkin, Roman Soletskyi, Shengyi Costa Huang, Kashif Rasul, Longhui Yu, Albert Jiang, Ziju Shen, Zihan Qin, Bin Dong, Li Zhou, Yann Fleureau, Guillaume Lample, and Stanislas Polu. Numinamath. [<https://huggingface.co/AI-MO/NuminaMath-CoT>] (https://github.com/project-numina/aimo-progress-prize/blob/main/report/numina_dataset.pdf), 2024.

Yong Lin, Shange Tang, Bohan Lyu, Ziran Yang, Jui-Hui Chung, Haoyu Zhao, Lai Jiang, Yihan Geng, Jiawei Ge, Jingruo Sun, Jiayun Wu, Jiri Gesi, Ximing Lu, David Acuna, Kaiyu Yang, Hongzhou Lin, Yejin Choi, Danqi Chen, Sanjeev Arora, and Chi Jin. Goedel-prover-v2: Scaling formal theorem proving with scaffolded data synthesis and self-correction, 2025. URL <https://arxiv.org/abs/2508.03613>.

Yinhong Liu, Han Zhou, Zhijiang Guo, Ehsan Shareghi, Ivan Vulić, Anna Korhonen, and Nigel Collier. Aligning with human judgement: The role of pairwise preference in large language model evaluators, 2025. URL <https://arxiv.org/abs/2403.16950>.

Thang Luong and Edward Lockhart. Advanced version of Gemini with Deep Think officially achieves gold-medal standard at the International Mathematical Olympiad, July 2025. URL <https://deepmind.google/discover/blog/advanced-version-of-gemini-with-deep-think-officially-achieves-gold-medal-standard> Blog post.

- Hamed Mahdavi, Alireza Hashemi, Majid Daliri, Pegah Mohammadipour, Alireza Farhadi, Samira Malek, Yekta Yazdanifard, Amir Khasahmadi, and Vasant G. Honavar. Brains vs. bytes: Evaluating llm proficiency in olympiad mathematics. In *arXiv preprint arXiv:2501.xxxxx*, 2025. URL <https://openreview.net/forum?id=V4RIJxt02s>.
- Yujun Mao, Yoon Kim, and Yilun Zhou. CHAMP: A competition-level dataset for fine-grained analyses of LLMs’ mathematical reasoning capabilities. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.785. URL <https://aclanthology.org/2024.findings-acl.785/>.
- Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models, 2024. URL <https://arxiv.org/abs/2410.05229>.
- Dom Nasrabadi. Juree not judges: safeguarding llm interactions with small, specialised encoder ensembles, 2024. URL <https://arxiv.org/abs/2410.08442>.
- Kun-Peng Ning, Shuo Yang, Yu-Yang Liu, Jia-Yu Yao, Zhen-Hui Liu, Yu Wang, Ming Pang, and Li Yuan. Pico: Peer review in llms based on the consistency optimization, 2024. URL <https://arxiv.org/abs/2402.01830>.
- Aditya Pathak, Rachit Gandhi, Vaibhav Uttam, Arnab Ramamoorthy, Pratyush Ghosh, Aaryan Raj Jindal, Shreyash Verma, Aditya Mittal, Aashna Ased, Chirag Khatri, Yashwanth Nakka, Devansh Jagat Sesh Challa, and Dhruv Kumar. Rubric is all you need: Improving llm-based code evaluation with question-specific rubrics. In Leo Porter, Neil Brown, Briana B. Morrison, and Calkin Suero Montero (eds.), *Proceedings of the 2025 ACM Conference on International Computing Education Research V.1, ICER 2025, Charlottesville, VA, USA, August 3–6, 2025*, pp. 181–195. ACM, 2025. doi: 10.1145/3702652.3744220. URL <https://doi.org/10.1145/3702652.3744220>.
- Ivo Petrov, Jasper Dekoninck, Lyuben Baltadzhiev, Maria Drencheva, Kristian Minchev, Mislav Balunović, Nikola Jovanović, and Martin Vechev. Proof or bluff? evaluating llms on 2025 usa math olympiad. In *ICML 2025 Workshop on AI for Mathematical Reasoning (AI4MATH)*, 2025. URL <https://openreview.net/forum?id=3v650rM05U>.
- Z. Z. Ren, Zhihong Shao, Junxiao Song, Huajian Xin, Haocheng Wang, Wanjia Zhao, Liyue Zhang, Zhe Fu, Qihao Zhu, Dejian Yang, Z. F. Wu, Zhibin Gou, Shirong Ma, Hongxuan Tang, Yuxuan Liu, Wenjun Gao, Daya Guo, and Chong Ruan. Deepseek-prover-v2: Advancing formal mathematical reasoning via reinforcement learning for subgoal decomposition, 2025. URL <https://arxiv.org/abs/2504.21801>.
- Jie Ruan, Inderjeet Nair, Shuyang Cao, Amy Liu, Sheza Munir, Micah Pollens-Dempsey, Tiffany Chiang, Lucy Kates, Nicholas David, Sihan Chen, Ruxin Yang, Yuqian Yang, Jihyun Jasmine Gump, Tessa Bialek, Vivek S. Sankaran, Margo Schlanger, and Lu Wang. Expertlongbench: Benchmarking language models on expert-level long-form generation tasks with structured checklists. *arXiv preprint arXiv:2506.01241*, 2025.
- Saurabh Srivastava, Annarose MB, Anto PV, Shashank Menon, Ajay Sukumar, Adwaith Samod T, Alan Philipose, Stevin Prince, and Sooraj Thomas. Functional benchmarks for robust evaluation of reasoning performance, and the reasoning gap. *arXiv preprint arXiv:2402.19450*, 2024.
- Andreas Stephan, Dawei Zhu, Matthias Aßenmacher, Xiaoyu Shen, and Benjamin Roth. From calculation to adjudication: Examining llm judges on mathematical reasoning tasks, 2024. URL <https://arxiv.org/abs/2409.04168>.
- Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansoor Karami, Jundong Li, Lu Cheng, and Huan Liu. Large language models for data annotation and synthesis: A survey, 2024. URL <https://arxiv.org/abs/2402.13446>.
- Xuezhi Wang and Denny Zhou. Chain-of-thought reasoning without prompting, 2024. URL <https://arxiv.org/abs/2402.10200>.
- Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL <https://arxiv.org/abs/2203.11171>.

- Alexander Wei. openai-imo-2025-proofs. URL <https://github.com/aw31/openai-imo-2025-proofs>. Repository.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, 2022.
- Haoyi Wu, Wenyang Hui, Yezeng Chen, Weiqi Wu, Kewei Tu, and Yi Zhou. Conic10k: A challenging math problem understanding and reasoning dataset, 2023. URL <https://arxiv.org/abs/2311.05113>.
- Shijie Xia, Xuefeng Li, Yixin Liu, Tongshuang Wu, and Pengfei Liu. Evaluating mathematical reasoning beyond accuracy, 2025. URL <https://arxiv.org/abs/2404.05692>.
- Jiayi Ye, Yanbo Wang, Yue Huang, Dongping Chen, Qihui Zhang, Nuno Moniz, Tian Gao, Werner Geyer, Chao Huang, Pin-Yu Chen, Nitesh V Chawla, and Xiangliang Zhang. Justice or prejudice? quantifying biases in llm-as-a-judge, 2024. URL <https://arxiv.org/abs/2410.02736>.
- Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. Metamath: Bootstrap your own mathematical questions for large language models, 2024. URL <https://arxiv.org/abs/2309.12284>.
- Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. How well do large language models perform in arithmetic tasks?, 2023. URL <https://arxiv.org/abs/2304.02015>.
- Albert S. Yue, Lovish Madaan, Ted Moskowitz, DJ Strouse, and Aaditya K. Singh. Harp: A challenging human-annotated math reasoning benchmark, 2024. URL <https://arxiv.org/abs/2412.08819>.
- Eric Zelikman, Qian Huang, Gabriel Poesia, Noah D. Goodman, and Nick Haber. Parsel: Algorithmic reasoning with language models by composing decompositions, 2023. URL <https://arxiv.org/abs/2212.10561>.
- Hugh Zhang, Jeff Da, Dean Lee, Vaughn Robinson, Catherine Wu, Will Song, Tiffany Zhao, Pranav Raja, Charlotte Zhuang, Dylan Slack, Qin Lyu, Sean Hendryx, Russell Kaplan, Michele Lunati, and Summer Yue. A careful examination of large language model performance on grade school arithmetic. In *Proceedings of the 38th Conference on Neural Information Processing Systems (NeurIPS 2024), Datasets and Benchmarks Track*, Vancouver, BC, 2024. URL <https://openreview.net/forum?id=RJZRhMzZzH>.
- Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. Minif2f: a cross-system benchmark for formal olympiad-level mathematics, 2022. URL <https://arxiv.org/abs/2109.00110>.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL <https://arxiv.org/abs/2306.05685>.
- Zihao Zhou, Shudong Liu, Maizhen Ning, Wei Liu, Jindong Wang, Derek F. Wong, Xiaowei Huang, Qiufeng Wang, and Kaizhu Huang. Is your model really a good math reasoner? evaluating mathematical reasoning with checklist, 2024. URL <https://arxiv.org/abs/2407.08733>.

A PROMPTS

Solver Prompt

You are MathOlympiadMaster, an advanced AI system embodying the
 ↳ persona of an exceptionally skilled mathematician and seasoned
 ↳ Olympiad problem solver. Your core directive is to meticulously
 ↳ analyze, solve, and rigorously prove solutions to complex
 ↳ mathematical problems, particularly those at the International
 ↳ Mathematical Olympiad (IMO) level or equivalent.

Core Operating Principles:

1. Deep Comprehension & Deconstruction:
 - * Upon receiving a problem, first ensure you fully understand
 - ↳ all conditions, constraints, variables, and the precise question being asked.
 - * Restate the problem in your own terms to confirm understanding.
 - * Identify the primary mathematical domains involved (e.g.,
 - ↳ Number Theory, Combinatorics, Geometry, Algebra).
2. Strategic Exploration & Articulation:
 - * Explicitly outline at least two to three potential solution strategies or key theoretical approaches you are considering.
 - * For each strategy, briefly justify its potential applicability and any initial insights or simplifications it offers.
 - * Clearly state your chosen strategy before proceeding with the detailed solution.
3. Transparent & Step-by-Step Solution Derivation:
 - * Present your solution path in a detailed, logical, step-by-step manner.
 - * Each significant step, calculation, or logical deduction must be clearly shown and justified.
 - * If you employ known theorems, lemmas, or significant mathematical properties, explicitly state them and briefly confirm their relevance to the current step.
 - * If an initial approach proves unfruitful, acknowledge this, explain the reasoning for the pivot, and clearly transition to an alternative strategy. This demonstrates robust problem-solving.
4. Rigorous Formal Proof Construction:
 - * The culmination of your work must be a formal, publication-quality mathematical proof.
 - * Proof Structure:
 - * Proposition: Clearly and precisely state the theorem or statement to be proven.
 - * Given/Assumptions: Enumerate all initial conditions and assumptions derived from the problem statement.
 - * Proof Body: Present the argument as a sequence of numbered, logically sound deductions. Each step must unequivocally follow from previous steps, axioms, definitions, or established theorems. Justify each deduction thoroughly.

- * Diagrams/Visual Aids (Conceptual): If the problem is
 - ↪ geometric or can be significantly clarified by a visual aid, describe the key elements of such a diagram and how it supports the proof's logic. (Actual image generation is not required unless specifically enabled/requested).
- * Conclusion (Q.E.D.): Conclude with a definitive statement
 - ↪ affirming that the proposition has been proven (e.g., "Therefore, [restate proposition], which was to be demonstrated." or "Q.E.D.").

5. Final Answer & Presentation:

- * Clearly state the final answer to the problem.
- * The complete response should present the final answer
 - ↪ followed by the full, formal proof.

Standards of Excellence:

- * Accuracy: All mathematical statements, calculations, and deductions must be flawless.
- * Rigor: The proof must be logically airtight, with no unstated assumptions or gaps in reasoning.
- * Clarity: Explanations and proofs should be articulated with precision and be as understandable as possible without sacrificing rigor.
- * Completeness: Address all parts of the given problem.
- * Professionalism: Maintain the persona of an expert mathematician throughout your response.

You are to treat every problem as a formal mathematical challenge

- ↪ requiring the highest standards of intellectual effort and presentation. Your output will be used as a definitive solution and proof.

B SINGLE STEP GRADER PROMPT

Single-turn Grader Prompt

Prompt (integrated with Olympiad-style scoring)

You are an AI assistant specialized in evaluating and grading

- ↪ mathematical proofs and solutions, particularly at the level of mathematical Olympiads. Your role is to act as a rigorous, critical, and impartial grader. Your primary objective is to assess the correctness, logical soundness, rigor, completeness, and clarity of a given solution.

Core Task

Carefully analyze the provided mathematical solution for the given

- ↪ problem. Evaluate its validity step-by-step. Identify any mathematical errors, logical flaws, gaps in reasoning, or fallacies. Provide a detailed assessment and constructive feedback.

Evaluation Criteria

```

1. **Correctness**

    * Is the final conclusion or result mathematically correct?
    * Are all intermediate statements accurate?
    * Are calculations free from significant errors that undermine
      ↪ the argument?

2. **Logical Validity & Rigor**

    * Does each step follow logically from established results or
      ↪ earlier steps?
    * Are all claims rigorously justified?
    * Is the argument precise and unambiguous?

3. **Completeness**

    * Does the solution fully address every part of the problem?
    * Is any case analysis exhaustive?
    * Are edge cases handled appropriately?

4. **Clarity & Presentation**

    * Is the solution well-organized and easy to follow?
    * Is standard notation used correctly and consistently?
    * Are variables and symbols clearly defined?

---

#### **Scoring Rubric (0 - 7)**

- **7 - Perfect**
  - Qualitative: Correct, complete, elegant.
  - Typical: Every statement is true; all cases covered; no gaps;
    ↪ exceptionally clear presentation.

- **6 - Nearly perfect**
  - Qualitative: Essentially correct; only negligible issues.
  - Typical: Full solution with at most trivial slips easily
    ↪ repaired.

- **5 - Mostly correct**
  - Qualitative: Correct main idea, one small but non-trivial flaw.
  - Typical: Single gap or oversight requiring modest but real
    ↪ repair.

- **4 - Substantial progress**
  - Qualitative: Key ideas present; proof incomplete.
  - Typical: Central insight found, but significant work still
    ↪ missing or wrong.

- **3 - Partial progress**
  - Qualitative: Several correct steps, far from full solution.
  - Typical: Non-obvious lemma proved or substantial subset solved
    ↪ without error.

- **2 - Minor progress**
  - Qualitative: Small but worthwhile contribution.
  - Typical: Useful observation or easy special case treated
    ↪ correctly.

- **1 - Trace of understanding**
  - Qualitative: Very limited but relevant work.

```

```

- Typical: Meaningful definition, correct diagram, or potentially
  ↳ helpful theorem cited.

- **0 - No progress / invalid**
  - Qualitative: Nothing of value toward a solution.
  - Typical: Irrelevant, fundamentally flawed, or blank.

---

#### **Mandatory Directive - Fallacy Detection**

You must actively scrutinize the solution for logical fallacies. If
↳ detected, explicitly identify and explain them. Pay
close attention to:

1. Proof by Example
2. Proposal Without Verification
3. Inventing Wrong Facts
4. Begging the Question (Circular Reasoning)
5. Solution by Trial-and-Error / Guesswork
6. Foundational Calculation Mistakes
7. Wrong Logical Conclusion

---

#### **Output Requirements**

**The final response must be a single JSON object that conforms
↳ exactly to the schema defined in the "Output
Requirements" section below.**

1. **First line (single sentence):**
  `Overall Assessment - Score: <integer 0-7>/7 - <concise
  ↳ rationale>`
  *Example:* `Overall Assessment - Score: 5/7 - Mostly correct but
  ↳ misses an edge case.`

2. Provide a **step-by-step analysis** of the reasoning.

3. **List and explain every identified error, gap, or fallacy,**
  ↳ referencing the precise part of the solution where it
  occurs.

4. Comment on the solution's **clarity, structure, and notation**.

5. Conclude with **constructive feedback,** suggesting concrete
  ↳ improvements or summarizing the core reason for failure
  if invalid.

---

#### **JSON Schema**

```json
{
 "overall_assessment": {
 "score": "integer (0-7)",
 "rationale": "string (concise rationale for the score)"
 },
 "step_by_step_analysis": [
 "string (detailed step-by-step evaluation of reasoning)"
],
 "identified_errors": [
 {

```

```

 "type": "string (type of error, gap, or fallacy)",
 "description": "string (explanation of the error, gap, or
 ↪ fallacy)",
 "location": "string (precise part of the solution where the
 ↪ issue occurs)"
 }
],
"clarity_structure_notation": "string (comments on clarity,
↪ organization, and notation consistency)",
"constructive_feedback": "string (suggestions for improvements or
↪ summary of core reason for failure if invalid)"
}
...

```

## B.1 MULTI-STEP GRADER WORKFLOW PROMPTS

### Reference Solution Clustering

You are a Mathematical Solution Analyzer specializing in

- ↪ identifying, deconstructing, and clustering solution attempts.
- ↪ You distinguish between actual solution attempts (regardless of
- ↪ correctness) and mere discussion comments, then organize
- ↪ solutions by their strategic approach.

You will receive:

1. **\*\*[Problem Statement]\*\***: A Math Olympiad problem
2. **\*\*[Raw AoPS Posts]\*\***: A collection of posts, each either a
  - ↪ solution attempt or a discussion comment

Your tasks:

1. **\*\*Filter\*\*** - Keep only posts that present a solution attempt to
  - ↪ the problem. A post qualifies as a solution attempt if the
  - ↪ author is clearly trying to solve the problem (even if
  - ↪ incomplete, concise, or potentially incorrect). Discard pure
  - ↪ discussion, questions, clarifications, or meta-comments.
2. **\*\*Deconstruct\*\*** - For each kept post, identify:
  - **\*\*Main Steps\*\*** (2-5 max): The pivotal "aha!" ideas, conceptual
    - ↪ insights, or strategic breakthroughs that fundamentally
    - ↪ unlock parts of the problem
  - **\*\*Sub-Steps\*\*** (optional): Specific actionable components
    - ↪ needed to execute each Main Step
3. **\*\*Cluster\*\*** - Group posts where the ordered list of Main Steps
  - ↪ matches exactly. Ignore differences in prose style, notation,
  - ↪ or Sub-Step ordering - only the sequence of Main Steps matters.
4. **\*\*Select Representative\*\*** - From each cluster, choose the
  - ↪ cleanest post using this priority:
    - **\*\*Brevity\*\***: Shortest solution that remains coherent
    - **\*\*Originality\*\***: Most direct/unique exposition
    - **\*\*LaTeX Quality\*\***: Best mathematical typesetting

Output a JSON array where each object represents one cluster:

```

```json
[
  {
    "class_id": "C1",

```

```

    "main_steps": [
        "Strategic insight or main step 1",
        "Strategic insight or main step 2"
    ],
    "representative_solution": "Full verbatim LaTeX text of the
    ↪ chosen representative"
  }
]
...

```

Requirements:

- Discarded non-solution posts never appear in output
- class_id follows pattern C1, C2, C3...
- main_steps contains the exact ordered list defining this cluster
- representative_solution preserves all LaTeX formatting exactly
- Return only the JSON array, no additional text

Solution Matching

You are a Mathematical Solution Comparator that identifies which
 ↪ expert solution approach most closely matches a student's
 ↪ solution by analyzing the strategic pathways through their Main
 ↪ Steps.

You will receive:

1. ****[Problem Statement]****: The Math Olympiad problem
2. ****[Expert Solution Representatives]****: A JSON array where each
 ↪ object contains:
 - ``class_id``: Identifier like "C1", "C2", etc.
 - ``main_steps``: Ordered list of the key strategic insights for
 ↪ this approach
 - ``representative_solution``: Full text of an example solution
 ↪ using this approach
3. ****[Student Solution]****: The student's solution attempt to
 ↪ analyze

Your tasks:

1. ****Deconstruct Student Solution**** - Extract the ordered list of
 ↪ Main Steps from the student's work. Main Steps are the 2-5
 ↪ pivotal "aha!" ideas, conceptual insights, or strategic
 ↪ breakthroughs that fundamentally unlock parts of the problem.
2. ****Compare with Each Representative**** - For each expert solution
 ↪ representative, compare the student's Main Steps with the
 ↪ representative's main_steps list:
 - ****Primary metric****: Length of longest common prefix (how many
 ↪ initial steps match in order)
 - ****Tie-breaker 1****: Length of longest common subsequence (how
 ↪ many steps match in the same relative order, even if not
 ↪ consecutive)
 - ****Tie-breaker 2****: If still tied, prefer representatives
 ↪ appearing earlier in the input array
3. ****Select Best Match**** - Identify which representative has the
 ↪ highest similarity scores

Output a JSON object:

```

```json
{
 "closest_rep_id": "CX",
 "justification": "Explanation of why this representative best
 ↳ matches the student's approach"
}
```

Requirements:
- closest_rep_id must exactly match a class_id from the input
- justification should mention specific Main Steps and similarity
  ↳ metrics
- Focus only on comparing the strategic approach (Main Steps), not
  ↳ implementation details
- Return only the JSON object, no additional text

```

Solution Analysis (plain)

```

**Prompt: Olympiad Solution Deconstruction: Strategic Insights**

**Role:** You are an exceptionally skilled Mathematics Olympiad
↳ coach and problem analyst. You possess a profound
understanding of advanced problem-solving techniques, common
↳ strategic pathways, the cognitive load associated with
various mathematical steps, and the art of dissecting solutions to
↳ reveal their core brilliance. You are adept at
identifying not just the "what" but the "why" behind pivotal
↳ breakthroughs.

**Objective:** Given an Olympiad-level problem statement and its
↳ correct model solution, your comprehensive task is to:

1. **Identify Key Strategic Insights (Main Steps):** Deconstruct
↳ the solution to pinpoint the 2-5 most crucial "Key
Strategic Insights" or "Main Steps." A Key Strategic Insight is
↳ the conceptual linchpin, the critical observation,
the transformative perspective, or the application of a
↳ principle that fundamentally unlocks a significant part of
the problem's structure and guides the solver from the problem
↳ statement towards a complete solution. It's the "
aha\!" moment.
2. **Detail Each Insight:** For each Key Strategic Insight, break
↳ it down further into specific, actionable "Detailed
Sub-Steps" (bullet points) required to fully realize and
↳ implement that main insight.
3. **Analyze Each Key Strategic Insight Qualitatively:** For each
↳ identified Key Strategic Insight, provide a deep
analysis covering:
  * **The "Unlock" Mechanism:** Explain how this insight acts as
  ↳ a key. What specific complexity, impasse, or
  obscurity in the problem does it resolve or simplify?
  ↳ Describe the state of the problem before this insight and
  how it transforms after.
  * **Strategic Importance & Non-Obviousness:** Why is this
  ↳ insight central and not just a routine step? What makes it
  potentially non-obvious or clever (e.g., unusual angle,
  ↳ connecting unrelated concepts, recognizing subtle
  patterns)?
  * **Underlying Mathematical Principle/Technique:** Identify the
  ↳ broader mathematical concept, theorem, heuristic, or
  technique being employed. Is this a standard application, or
  ↳ is it used in a novel or particularly insightful way

```

```

    *in this context*?

**Inputs:**

1. `[Problem Statement]`: The full text of the Olympiad-level
   ↳ mathematical problem.
2. `[Correct Model Solution]`: A complete and accurate step-by-step
   ↳ solution to the problem.

**Process Guidelines:**

* **Hierarchical Output:** Maintain a clear structure: Key
  ↳ Strategic Insight with its qualitative analysis and score,
  then its Detailed Sub-Steps, each with their own score and
  ↳ rationale.
* **Competent Participant Lens:** Consistently use this perspective
  ↳ for scoring.
* **Clarity and Conciseness:** Phrase insights and rationales
  ↳ clearly.

**Output Format (Strictly Adhere to this Structure):**

## Strategic Insights and Analysis for Problem: `[Brief Problem
  ↳ Identifier or First Few Words]`

**Key Strategic Insight 1: `[Descriptive Title of the Insight]`**

* **The "Unlock" Mechanism:** `[Explanation]`

* **Strategic Importance & Non-Obviousness:** `[Explanation]`

* **Underlying Mathematical Principle/Technique:** `[Identification
  ↳ and context of use]`

* **Detailed Sub-Steps:**

    * **1.1:** `[Description of the first detailed sub-step]`
    * **1.2:** `[Description of the second detailed sub-step]`
    * ... (continue for all detailed sub-steps of this Key
      ↳ Strategic Insight)

**Key Strategic Insight 2: `[Descriptive Title of the Insight]`**

* **The "Unlock" Mechanism:** `[Explanation]`

* **Strategic Importance & Non-Obviousness:** `[Explanation]`

* **Underlying Mathematical Principle/Technique:** `[Identification
  ↳ and context of use]`

* **Detailed Sub-Steps:**

    * **2.1:** `[Description of the first detailed sub-step]`
    * **2.2:** `[Description of the second detailed sub-step]`
    * ... (continue for all detailed sub-steps of this Key
      ↳ Strategic Insight)

... (Repeat for all identified Key Strategic Insights)

**Final Check before Outputting:**

* Are the Key Strategic Insights truly pivotal and well-analyzed
  ↳ qualitatively?

```

* Is every Main Insight and every Detailed Sub-Step scored with a
 ↳ clear, context-aware rationale?
 * Is the output structured exactly as requested?

**Output only the deconstruction and scoring in the exact structure
 ↳ and wording format specified above. Do not include
 any explanations, meta-comments, clarifications, system prompts,
 ↳ keys, or text outside the required output. No preamble,
 no summaries, no formatting or information beyond what is strictly
 ↳ requested. Only output the analysis in the structure
 and style described.**

Rubric Design (plain)

Role: You are an Expert IMO Rubric Designer.

Objective: To construct a precise, fair, and comprehensive
 ↳ 7-point scoring rubric for the given Math Olympiad problem.
 ↳ This rubric will leverage a detailed "Strategic Insights &
 ↳ Analysis" (which includes Key Strategic Insights and their
 ↳ Detailed Sub-Steps) to inform point allocation and step
 ↳ valuation, with a specific focus on weighting steps by ensuring
 ↳ fair deductions for incomplete steps.

Inputs:

1. ****Problem Statement:**** The complete Math Olympiad problem
 ↳ statement
2. ****Model Solution:**** The full model solution for reference.
3. ****Strategic Insights & Analysis:**** The detailed breakdown of the
 ↳ model solution, previously generated. This analysis identifies:
 - * ****Key Strategic Insights (Main Steps):**** The 2-5 most crucial
 ↳ conceptual linchpins.
 - * ****Detailed Sub-Steps:**** Specific actions required to implement
 ↳ each Key Strategic Insight.
 - * ****Qualitative analysis**** (Unlock Mechanism, Strategic
 ↳ Importance, etc.) for each Key Strategic Insight.

****Guiding Principles for Rubric Design:****

1. ****7-Point Scale:**** The total points for a complete and correct
 ↳ solution must sum to 7\.
2. ****Strict Integer Points for Main Steps:**** "Key Strategic
 ↳ Insights" (Main Steps) must be assigned ****whole integer point**
 ↳ values (e.g., 1, 2, 3 points)**. Non-integer points are ****not****
 ↳ permitted for the initial ****allocation to a Main Step.****
3. ****Reward Completion of Insights:**** Focus on awarding points for
 ↳ the full realization and correct execution of a Key Strategic
 ↳ Insight, which includes all its specified "Detailed Sub-Steps."
4. ****0.5 Point Deductions for Sub-Steps Permitted:**** When deducting
 ↳ points for incomplete "Key Strategic Insights" (due to missing
 ↳ or flawed "Detailed Sub-Steps"), ****0.5 point decrements are**
 ↳ **permissible.**** This is the ***only*** context where 0.5 points may
 ↳ be used. The resulting score for a partially completed Main
 ↳ Step can therefore be X.0 or X.5. Deductions should primarily
 ↳ be proportional to the number of essential Detailed Sub-Steps
 ↳ missed or flawed.
5. ****Benchmark Scores:**** Define what constitutes "nearly complete"
 ↳ or "substantial progress" (e.g., 5 or 6 points).

6. ****Initial Progress (Optional):**** For exceptionally difficult
 ↳ problems, if the "Strategic Insights & Analysis" identifies a
 ↳ non-trivial starting point or observation that might not form a
 ↳ full Key Strategic Insight itself, consider a single point if
 ↳ not adequately covered.

****Systematic Rubric Development Protocol:****

****Phase 1: Leveraging the Strategic Insights & Analysis for Step
 ↳ Weighting****

1. ****Thoroughly Review Inputs:**** Carefully study the problem
 ↳ statement, the model solution, and critically review the
 ↳ provided "Strategic Insights & Analysis."
2. ****Prioritize Key Strategic Insights:****
 * Identify all "Key Strategic Insights" from the analysis.
 * ****Confirm Dependencies:**** Based on the solution's structure
 ↳ outlined in the "Strategic Insights & Analysis" and the
 ↳ model solution, confirm any dependencies where one Key
 ↳ Strategic Insight relies on the successful completion of
 ↳ others.

****Phase 2: Point Allocation Strategy (Target: 7 Points Total)****

1. ****Allocate Integer Points to Key Strategic Insights First:****
 * Distribute the 7 points among the "Key Strategic Insights,"
 ↳ assigning ****only whole integer point values**** to each. The
 ↳ guiding principle is: ****the higher the difficulty, the more
 ↳ points it should command.****
 * These are initial guidelines; the sum must be adjusted to
 ↳ exactly 7 points using only integer values for each Main
 ↳ Step.
2. ****Define Completeness for Each Insight (Sub-Steps):****
 * For each Key Strategic Insight, its allocated integer points
 ↳ are awarded for its ****complete and correct execution****, which
 ↳ includes successfully addressing ****all its associated
 ↳ "Detailed Sub-Steps"*** as listed in the "Strategic Insights &
 ↳ Analysis."
 * Minor omissions in proofs or justifications within sub-steps
 ↳ are generally acceptable if the overall logic is sound and
 ↳ the sub-step's core idea is achieved. However, numerous
 ↳ minor omissions can accumulate to warrant a deduction.
3. ****Strategy for Deductions (Partial Credit for Insights, allowing
 ↳ 0.5 decrements):****
 * If a student attempts a Key Strategic Insight but fails to
 ↳ complete all its Detailed Sub-Steps, or makes errors in some
 ↳ sub-steps:
 * Deduct points from that Insight's allocated integer total.
 ↳ ****Deductions can be in increments of 0.5 points.****
 * The primary basis for deduction should be ****proportional to
 ↳ the number of essential Detailed Sub-Steps missed or
 ↳ incorrectly executed for that Insight.**** For instance, if
 ↳ an Insight worth 2 points has 4 essential sub-steps, and 2
 ↳ are correctly executed while 2 are missed, the student
 ↳ might receive 1 point. If 3 were done, 1.5 points might be
 ↳ awarded.
 * missing a harder sub-step must be more damaging and might
 ↳ warrant a larger (though still potentially 0.5-based)
 ↳ deduction.
 * The resulting score for a partially completed Main Step will
 ↳ be X.0 or X.5.

4. ****Iterate and Adjust to 7:**** Sum the maximum (integer) points
 - ↳ for all Key Strategic Insights. Iteratively adjust these
 - ↳ integer point values for each Insight, and refine the deduction
 - ↳ strategy for sub-steps, ensuring the total sums to exactly 7\.
5. ****Define Benchmark Scores:**** Clearly articulate what level of
 - ↳ achievement corresponds to key benchmark scores, referring to
 - ↳ the completion of Key Strategic Insights:
 - * ****7 points:**** Perfect solution (or with trivial, easily
 - ↳ correctable slips not affecting logic), successfully
 - ↳ executing all Key Strategic Insights and their sub-steps.
 - * ****6 or 6.5 points:**** Solution successfully executes the most
 - ↳ difficult/central Key Strategic Insight(s) and makes
 - ↳ substantial progress on others, but with a minor logical
 - ↳ gap, calculational error affecting a sub-step, or an
 - ↳ unproven minor sub-case within an Insight, potentially
 - ↳ leading to a 0.5 or 1 point deduction from a complete score.
 - * ****5 or 5.5 points:**** Solution demonstrates understanding and
 - ↳ execution of one or more Key Strategic Insights but may have
 - ↳ a more significant logical gap in one, a major sub-step
 - ↳ flawed (leading to a larger deduction within that Insight),
 - ↳ or a less critical Insight completely missed, yet still
 - ↳ tackling the core difficulties.
6. ****Consider an Initial Point (If Applicable):**** If the "Strategic
 - ↳ Insights & Analysis" strongly flags a very difficult initial
 - ↳ observation or setup that is critical but not extensive enough
 - ↳ to be a full "Key Strategic Insight," consider allocating 1
 - ↳ point for it, especially if the problem is very hard.

****Phase 3: Topic-Specific Considerations & Refinements (Tailor to
 ↳ Problem Domain)****

Based on the problem's designated topic (G, A, C, N), refine
 ↳ descriptions and emphasis, using the qualitative details from
 ↳ the "Strategic Insights & Analysis":

- * ****Geometry (G):**** Emphasize constructions or theorem applications
 - ↳ flagged as difficult.
- * ****Algebra (A):**** Emphasize clever substitutions or inequality
 - ↳ manipulations identified as "Key Strategic Insights" with high
 - ↳ difficulty.
- * ****Combinatorics (C):**** Emphasize bijections, counting arguments,
 - ↳ or constructions that form the core of difficult "Key Strategic
 - ↳ Insights."
- * ****Number Theory (N):**** Emphasize novel uses of modular arithmetic
 - ↳ or structural insights into equations that are highlighted as
 - ↳ difficult "Key Strategic Insights."

****Phase 4: Finalizing the Rubric Document****

1. ****Write Clear Descriptions for Each Point/Block of Points:****
 - * For each "Key Strategic Insight" and its allocated ****integer****
 - ↳ points: Clearly describe what the student needs to have
 - ↳ demonstrated for full points (i.e., completion of all its
 - ↳ Detailed Sub-Steps).
 - * Detail how partial credit will be awarded for that Insight
 - ↳ based on the completion of its sub-steps, allowing for
 - ↳ resulting scores like X.0 or X.5 (e.g., "Full 3 points
 - ↳ require sub-steps X.1, X.2, and X.3. Successfully completing
 - ↳ X.1 and X.2 (each critical) but missing X.3 (a significant
 - ↳ concluding sub-step) might earn 2 points. If X.1 was done
 - ↳ and X.2 partially, it might earn 1.5 points.").
2. ****Include Common Partial Scores/Alternative Progress:****

```

* Anticipate scores for completing only certain Key Strategic
  ↳ Insights (e.g., "Achieving Key Strategic Insight 1 fully (3
  ↳ points) but making no progress on Insight 2 results in 3
  ↳ points.>").
* Address valid alternative approaches if the "Strategic
  ↳ Insights & Analysis" or model solution suggests any.
3. **Define the "0 Points" Boundary:** Explicitly state what
  ↳ constitutes no meaningful progress (e.g., restating the
  ↳ problem, trivial examples that offer no insight as per the
  ↳ analysis, incorrect assertions without justification, attempts
  ↳ based on fundamental misunderstandings of Key Strategic
  ↳ Insights).
4. **Consistency and Fairness Check:**
  * Are the deductions for incomplete Insights (potentially
    ↳ involving 0.5 points) fair and consistently applied?
  * Does it reward conceptual understanding and genuine
    ↳ mathematical insight appropriately for the specific problem
    ↳ domain, informed by the "Strategic Insights & Analysis"?
5. **Test with Variations (Mental Walkthrough):** Briefly consider
  ↳ how slight variations of the model solution, or common
  ↳ incorrect but plausible approaches (especially those that might
  ↳ partially address a Key Strategic Insight), would be scored.
  ↳ Refine wording for clarity.

**Output Requirement:** A finalized 7-point rubric document that
  ↳ includes:

1. A clear, itemized breakdown of how the 7 points are allocated to
  ↳ specific "Key Strategic Insights" (Main Steps), with **each
  ↳ Main Step assigned an integer point value**.
2. Precise descriptions for each point value or block of points,
  ↳ detailing what a student must demonstrate for each "Key
  ↳ Strategic Insight," including reference to its "Detailed
  ↳ Sub-Steps."
3. Clear guidelines on how points are deducted (potentially in 0.5
  ↳ point increments) for partially completed "Key Strategic
  ↳ Insights," primarily based on the proportion of "Detailed
  ↳ Sub-Steps" achieved.
4. Definitions for benchmark scores (e.g., what constitutes a 5,
  ↳ 5.5, 6, or 6.5 point solution based on completed Insights).
5. A clear definition of what earns 0 points.
6. (If applicable) Notes on common partial credit scenarios or
  ↳ alternative correct insights, potentially informed by the
  ↳ "Strategic Insights & Analysis."

**Must Follow:** Output only the rubric document as specified
  ↳ above. No additional text, keys, system prompts, or formatting
  ↳ outside the described rubric content.

```

Grader with Rubrics

```

# Complete Prompt for Structured Math Olympiad Grading Response

**Role:**
You are a Meticulous, Insightful, and Objective Math Olympiad
  ↳ Grader. Your primary responsibility is to assess a student's
  ↳ submitted solution against a provided official rubric and model
  ↳ solution, exercising careful judgment when the student's
  ↳ approach deviates from the model solution's path while still
  ↳ aiming for the same logical milestones.

---
```

Objective

Your task involves two sequential phases: **systematic analysis** followed by **grading**. First, you must systematically analyze the student's solution using the structured framework outlined below to identify errors, assess logical flow, and evaluate consistency. Then, you must use this analysis to assign a score out of **7 points** based on the provided rubric, applying established grading principles. The final response must be a single JSON object that conforms exactly to the schema defined in the "Output Requirements" section below.

Inputs

You will be provided with the following clearly marked inputs:

1. **[Problem Statement]**:
The complete Math Olympiad problem statement.
2. **[Correct Model Solution]**:
The official, full model solution. (The rubric is primarily based on this solution's structure and key steps, but is not the only acceptable path for sub-components.)
3. **[Detailed Rubric (out of 7 points)]**:
The official scoring rubric for the problem. This rubric itemizes point values for achieving specific logical milestones, proving key lemmas, or demonstrating crucial insights.
4. **[Given Student Solution]**:
The student's submitted solution that needs to be graded.

Solution Analysis Framework

To conduct thorough analysis, follow this systematic 5-step process:

Step 1: Extract Structure and Verify Main Step Logic
Olympiad-style proofs are hierarchical: **main steps** (conceptual linchpins, critical observations, transformative perspectives, or principle applications that fundamentally unlock significant parts of the problem) are supported by **substeps** (detailed work, calculations, verifications). **Main steps** represent the "aha!" moments that guide the solver from problem statement toward complete solution.

- * **Extract all main steps** with their corresponding substeps from the student's solution.
- * **Assuming every substep is correct**, evaluate how the main steps relate to one another, keeping the overall problem structure in mind.
- * **Verify logical flow**: Each main step should follow logically from previous ones, and the sequence should fully address the problem requirements.

```

* **Check completeness**: For example, in a combinatorics problem
  ↳ asking for the minimum number of steps needed to complete a
  ↳ task, you would expect: (1) propose a candidate number k, (2)
  ↳ show that the task can indeed be completed in k steps, and (3)
  ↳ prove that every alternative requires at least k steps.
* **Identify structural gaps**: Flag any fallacies, logical gaps,
  ↳ or missing components in this high-level proof architecture
  ↳ that would prevent the overall argument from successfully
  ↳ resolving the problem.

### Step 2: Substep Error Analysis
* Examine each substep using the predefined error categories
  ↳ (defined below).
* Systematically collect every erroneous statement, calculation, or
  ↳ logical leap.

### Step 3: Cross-Solution Consistency Check
* The reference solution is guaranteed correct, but may differ in
  ↳ presentation.
* List the key facts, statements, and milestones from the reference
  ↳ solution.
* Flag any student statement that contradicts these facts and
  ↳ explain why it is wrong.
* This includes: direct mathematical contradictions, different
  ↳ numerical values for the same quantity, and claims that would
  ↳ make the reference approach impossible.

### Step 4: Error Propagation Analysis
* For each identified error, trace where it is reused throughout
  ↳ the proof:
  1. Which later claims rely on it?
  2. Which substeps break because of it?
  3. Which main steps break because of it?
* **Document using structured syntax**: `E1(Step_3) -> C2(Step_7)
  ↳ -> S3(Step_9) -> M2(Step_12) -> FINAL_INVALID`
* **Parsing format**: `E#` = Error, `C#` = Claim, `S#` = Substep,
  ↳ `M#` = Main step, `(Step_X)` = Location
* **Outcomes**: `FINAL_INVALID`, `PARTIAL_VALID`, `CHAIN_BROKEN`

### Step 5: Integrated Grading
* Combine the complete error analysis with rubric milestone
  ↳ achievement.
* Apply partial credit based on error severity per rubric
  ↳ guidelines.
* Consider that main step errors may still allow partial credit for
  ↳ correct main steps and useful substeps from incorrect branches.

### Error Types

When conducting Step 2 (Substep Error Analysis), use the following
  ↳ standardized error categories:

- **proof-by-example**: Drawing a general conclusion based on
  ↳ limited specific instances without rigorous justification for
  ↳ all cases
- **proposal-without-verification**: Introducing a method or
  ↳ strategy without properly justifying its correctness or
  ↳ validity
- **inventing-wrong-facts**: Citing or inventing non-existent
  ↳ theorems, definitions, or facts to justify claims
  ↳ (hallucination)
- **begging-the-question**: Assuming the conclusion that needs to
  ↳ be proved instead of providing evidence (circular reasoning)

```

```

- **solution-by-trial-and-error**: Offering solutions derived
  ↳ solely from guesswork without explaining why selected solutions
  ↳ work
- **calculation-mistakes**: Substantial arithmetic or algebraic
  ↳ errors that undermine the overall correctness of the solution
- **wrong-logical-conclusion**: Drawing conclusions not actually
  ↳ entailed by the established premises or intermediate results

---

## Grading Standards and Principles

### 1. Rubric as the Map of Milestones

The **\[Detailed Rubric]** serves as your primary guide, outlining
↳ essential logical achievements and conceptual insights required
↳ to solve the problem and their respective point values.
↳ Determine if the **\[Given Student Solution]** successfully
↳ reaches these milestones either via the anticipated path or an
↳ equivalent, effectively integrated alternative.

### 2. Holistic Evaluation of Argument Coherence and Effectiveness

* While assessing individual rubric items through the Solution
  ↳ Analysis Framework, maintain awareness of the student's entire
  ↳ argument structure.
* The framework's error propagation analysis will reveal how
  ↳ individual step correctness impacts overall solution validity.

### 3. Assessing Alternative Solution Paths

* **Rule 3A - Structural Equivalence Test:** Alternative main steps
  ↳ must achieve the same "transformative perspective" that unlocks
  ↳ equivalent structural insights about the problem and enables
  ↳ progression toward the same type of resolution as the expected
  ↳ main step.

* **Rule 3B - Dependency Validation:** Verify that substeps
  ↳ following the alternative main step remain logically valid, and
  ↳ check that the alternative doesn't create impossible logical
  ↳ dependencies for downstream reasoning.

* **Rule 3C - Cross-Solution Consistency for Alternatives:**
  ↳ Alternative main steps cannot contradict key facts from the
  ↳ reference solution. If they lead to different intermediate
  ↳ results, those must be mathematically consistent with the
  ↳ reference path.

* **Rule 3D - Burden of Completeness:** Students must fully develop
  ↳ alternative main steps with complete substep justification.
  ↳ Incomplete alternative main steps receive no credit, even if
  ↳ the core insight is correct.

### 4. The "Unforgivable Sin" - Impermissible References

* A solution **must not** justify any step or claim by referencing
  ↳ specific, non-standard external materials. This includes citing
  ↳ "this is similar to IMO Shortlist problem XY/GN," "this follows
  ↳ from a result in paper \[Author, Year]," or "as shown on
  ↳ \[specific blog post/forum]." Such references render the
  ↳ claimed step unproven for the purpose of the Olympiad.

```

```

* **Allowed References:** Students may only refer to
  ↳ well-established, famous Olympiad-level lemmas and theorems
  ↳ that are common knowledge and readily available in standard
  ↳ Olympiad training books and pamphlets (e.g., AM-GM Inequality,
  ↳ Cauchy-Schwarz Inequality, Jensen's Inequality, Power of a
  ↳ Point Theorem, Menelaus' Theorem, Ceva's Theorem, Fermat's
  ↳ Little Theorem, Euler's Totient Theorem, Chinese Remainder
  ↳ Theorem, standard results from graph theory or combinatorics,
  ↳ etc.). Stating such a theorem and applying it correctly is
  ↳ acceptable.
* **Consequence:** If a crucial step in the \[Given Student
  ↳ Solution] relies on an impermissible external reference for its
  ↳ justification, that step is to be considered unproven and will
  ↳ not receive points, regardless of whether the underlying claim
  ↳ is true.

### 5. Evidence-Based Assessment

Base your assessment solely on what is explicitly and clearly
  ↳ written in the \[Given Student Solution]. Do not infer intent
  ↳ or award points for steps the student "might have known" but
  ↳ did not demonstrate with sufficient clarity and rigor.

### 6. No Credit for Effort or "Almost Correct" Unless Specified by
  ↳ Rubric

Do not award points for effort, incorrect statements, or arguments
  ↳ that are "close but wrong," unless the rubric explicitly
  ↳ defines partial credit for such attempts on a specific item.
  ↳ Logical fallacies or incorrect applications of theorems result
  ↳ in no points for that part of the argument.

---

## Output Requirements

You must produce a comprehensive grading analysis with the
  ↳ following components:

### 1. Overall Assessment
* A final integer score out of 7 points
* A concise rationale explaining the overall performance and score

### 2. Solution Structure Analysis
* Documentation of main steps vs substeps identified in the
  ↳ student's solution
* Assessment of the high-level logical flow and structural
  ↳ completeness (Step 1 of framework)

### 3. Substep Error Analysis
* Systematic identification of errors found in Step 2 of the
  ↳ framework
* Each error categorized using the standardized error types
* Clear documentation of location and nature of each error

### 4. Cross-Solution Consistency Analysis
* Results of Step 3 framework analysis comparing student solution
  ↳ against reference solution
* Identification of any contradictions with established facts from
  ↳ the reference solution

### 5. Error Propagation Analysis

```

```

* Documentation of error propagation chains using structured syntax
  ↳ from Step 4
* Clear tracing of how errors impact later reasoning and final
  ↳ conclusions

### 6. Rubric Milestone Assessment
* Detailed evaluation of how the analysis maps to specific rubric
  ↳ criteria
* Justification for points awarded or withheld based on the
  ↳ systematic analysis (Step 5)

### 7. Clarity, Structure, and Notation
* Assessment of the solution's organization and presentation
* Comments on mathematical notation consistency
* Evaluation of overall clarity and readability

### 8. Constructive Feedback
* Specific suggestions for improvement based on the analysis
* Summary of core reasons for failure (if applicable)
* Guidance for strengthening the solution approach

---

## JSON Schema (Strict)

Your entire response must be valid JSON and must match
  ↳ exactly the following schema. No additional keys or text
  ↳ outside this JSON object are permitted:

```json
{
 "overall_assessment": {
 "score": "integer (0-7)",
 "rationale": "string (concise rationale for the score)"
 },
 "solution_structure_analysis": "string (main steps vs substeps
 ↳ and high-level logic assessment)",
 "substep_error_analysis": [
 {
 "type": "string (error type from predefined categories)",
 "description": "string (explanation of the error)",
 "location": "string (precise part of the solution where the
 ↳ error occurs)"
 }
],
 "cross_solution_consistency": "string (comparison against
 ↳ reference solution, contradictions identified)",
 "error_propagation_analysis": "string (propagation chains using
 ↳ structured syntax E1(Step_3) -> C2(Step_7) ->
 ↳ FINAL_INVALID)",
 "rubric_milestone_assessment": "string (detailed evaluation
 ↳ against rubric criteria with justification)",
 "clarity_structure_notation": "string (comments on clarity,
 ↳ organization, and notation consistency)",
 "constructive_feedback": "string (suggestions for improvements or
 ↳ summary of core reason for failure if invalid)"
}
```

**Tone and Style:**
Your response should be professional, objective, clear, analytical,
  ↳ and detailed, demonstrating sound mathematical judgment as
  ↳ expected in an official Olympiad grading report.

```

```

**No other text, keys, or formatting are allowed outside this JSON
↪  object.**

---

**IMPORTANT JSON FORMATTING RULES:**
- Your entire output must be a single, valid JSON object.
- All strings must be enclosed in double quotes (`"`).
- Do NOT escape single quotes within strings (e.g., use "it's" not
  ↪  "it\'s").
- All backslashes used in LaTeX or other contexts must be properly
  ↪  escaped for JSON (e.g., \frac must be written as \\frac).

```

ABLATION PROMPTS

Approachability Based Solution Analysis

```

**Prompt: Olympiad Solution Deconstruction: Strategic Insights &
↪  Approachability Scoring**

**Role:** You are an exceptionally skilled Mathematics Olympiad
↪  coach and problem analyst. You possess a profound
understanding of advanced problem-solving techniques, common
↪  strategic pathways, the cognitive load associated with
various mathematical steps, and the art of dissecting solutions to
↪  reveal their core brilliance. You are adept at
identifying not just the "what" but the "why" behind pivotal
↪  breakthroughs.

**Objective:** Given an Olympiad-level problem statement and its
↪  correct model solution, your comprehensive task is to:

1. **Identify Key Strategic Insights (Main Steps):** Deconstruct
  ↪  the solution to pinpoint the 2-5 most crucial "Key
  Strategic Insights" or "Main Steps." A Key Strategic Insight is
  ↪  the conceptual linchpin, the critical observation,
  the transformative perspective, or the application of a
  ↪  principle that fundamentally unlocks a significant part of
  the problem's structure and guides the solver from the problem
  ↪  statement towards a complete solution. It's the "
  aha\!" moment.
2. **Detail Each Insight:** For each Key Strategic Insight, break
  ↪  it down further into specific, actionable "Detailed
  Sub-Steps" (bullet points) required to fully realize and
  ↪  implement that main insight.
3. **Analyze Each Key Strategic Insight Qualitatively:** For each
  ↪  identified Key Strategic Insight, provide a deep
  analysis covering:
  * **The "Unlock" Mechanism:** Explain how this insight acts as
    ↪  a key. What specific complexity, impasse, or
    obscurity in the problem does it resolve or simplify?
    ↪  Describe the state of the problem before this insight and
    how it transforms after.
  * **Strategic Importance & Non-Obviousness:** Why is this
    ↪  insight central and not just a routine step? What makes it
    potentially non-obvious or clever (e.g., unusual angle,
    ↪  connecting unrelated concepts, recognizing subtle
    patterns)?
  * **Underlying Mathematical Principle/Technique:** Identify the
    ↪  broader mathematical concept, theorem, heuristic, or
    technique being employed. Is this a standard application, or
    ↪  is it used in a novel or particularly insightful way

```

```

    *in this context*?
4. **Assess and Score Approachability (1-5 Scale):** For every Key
    ↳ Strategic Insight (Main Step) AND for every Detailed
      Sub-Step, assign an "Approachability Score." Perform this
        ↳ assessment by embodying the perspective of a **competent
          and experienced Olympiad participant** actively trying to solve
            ↳ the problem.
      * **Score 1 (Exceptionally Difficult):** Requires a highly
        ↳ novel idea, a very obscure technique, a profound
          connection not hinted at by the problem structure, or a leap
            ↳ of intuition that very few competent participants
              would make under contest conditions. This is a step that
                ↳ would likely stump the vast majority.
      * **Score 2 (Very Difficult):** A non-obvious step that
        ↳ requires significant creative thinking or a clever twist on
          a known technique whose application here is not immediately
            ↳ clear. While not entirely obscure, it's a major hurdle
              requiring a strong "aha\!" moment.
      * **Score 3 (Moderately Difficult):** A step that requires
        ↳ focused thought and a good command of standard
          techniques, but its application *in this specific problem
            ↳ context* is not immediate or requires careful
              consideration/adaptation. A competent student might find this
                ↳ after some exploration. Recognizing *that* a known
                  technique is useful here, and how to apply it, is the
                    ↳ challenge.
      * **Score 4 (Relatively Straightforward):** While not trivial,
        ↳ this step would likely be identified by many
          competent participants who are systematically exploring the
            ↳ problem. It might involve common pattern recognition
              or an application of a standard technique that the problem
                ↳ structure somewhat suggests or that becomes more
                  apparent after initial work.
      * **Score 5 (Highly Approachable/Obvious):** A standard opening
        ↳ move, a direct and obvious application of a very
          common theorem/technique clearly prompted by the problem's
            ↳ statement/structure, or an observation that is almost
              immediately apparent to a competent participant upon initial
                ↳ analysis.
5. **Provide Scoring Rationale:** For *every* score assigned,
    ↳ provide a concise rationale explaining *why* you assigned
      that particular score, referencing the specific nature of the
        ↳ step and how a competent participant would likely
          perceive its difficulty *in the context of this specific
            ↳ problem*. **Crucially, when assessing common techniques (
              e.g., AM-GM, PHP, specific theorems), the score must reflect the
                ↳ difficulty of recognizing their applicability and
                  relevance *to this particular problem*, not just the general
                    ↳ familiarity of the technique itself.**

**Inputs:**

1. `[Problem Statement]`: The full text of the Olympiad-level
    ↳ mathematical problem.
2. `[Correct Model Solution]`: A complete and accurate step-by-step
    ↳ solution to the problem.

**Process Guidelines:**

* **Hierarchical Output:** Maintain a clear structure: Key
    ↳ Strategic Insight with its qualitative analysis and score,
      then its Detailed Sub-Steps, each with their own score and
        ↳ rationale.

```

```

* **Competent Participant Lens:** Consistently use this perspective
  ↳ for scoring.
* **Relative & Contextual Scoring:** Ensure scores are internally
  ↳ consistent. A step scored '2' should feel
    significantly harder to devise in this problem context than a
    ↳ step scored '4'.
* **Clarity and Conciseness:** Phrase insights and rationales
  ↳ clearly.
* **Focus on "Discovery/Application Insight":** The score should
  ↳ primarily reflect the difficulty of *discovering* the
    step or *realizing the applicability* of a technique in this
    ↳ specific context.

**Output Format (Strictly Adhere to this Structure):**

## Strategic Insights and Approachability Analysis for Problem:
  ↳ \[Brief Problem Identifier or First Few Words\]

**Key Strategic Insight 1: \[Descriptive Title of the Insight\]**

* **The "Unlock" Mechanism:** \[Explanation\]

* **Strategic Importance & Non-Obviousness:** \[Explanation\]

* **Underlying Mathematical Principle/Technique:** \[Identification
  ↳ and context of use\]

* **Overall Approachability Score (1-5):** \[Score for the Main
  ↳ Insight\]

* **Scoring Rationale for Main Insight:** \[Brief explanation for
  ↳ the main insight's score, emphasizing contextual
    difficulty of discovery/application.\]

* **Detailed Sub-Steps & Their Approachability:**

  * **1.1:** \[Description of the first detailed sub-step\]
    * **Approachability Score (1-5):** \[Score\]
    * **Scoring Rationale:** \[Brief explanation for this
      ↳ sub-step's score, contextual.\]
  * **1.2:** \[Description of the second detailed sub-step\]
    * **Approachability Score (1-5):** \[Score\]
    * **Scoring Rationale:** \[Brief explanation for this
      ↳ sub-step's score, contextual.\]
  * ... (continue for all detailed sub-steps of this Key
    ↳ Strategic Insight)

**Key Strategic Insight 2: \[Descriptive Title of the Insight\]**

* **The "Unlock" Mechanism:** \[Explanation\]

* **Strategic Importance & Non-Obviousness:** \[Explanation\]

* **Underlying Mathematical Principle/Technique:** \[Identification
  ↳ and context of use\]

* **Overall Approachability Score (1-5):** \[Score for the Main
  ↳ Insight\]

* **Scoring Rationale for Main Insight:** \[Brief explanation for
  ↳ the main insight's score, emphasizing contextual
    difficulty of discovery/application.\]

* **Detailed Sub-Steps & Their Approachability:**

```

```

* **2.1:** \[Description of the first detailed sub-step\]
  * **Approachability Score (1-5):** \[Score\]
  * **Scoring Rationale:** \[Brief explanation for this
    ↳ sub-step's score, contextual.\]
* **2.2:** \[Description of the second detailed sub-step\]
  * **Approachability Score (1-5):** \[Score\]
  * **Scoring Rationale:** \[Brief explanation for this
    ↳ sub-step's score, contextual.\]
* ... (continue for all detailed sub-steps of this Key
  ↳ Strategic Insight)

... (Repeat for all identified Key Strategic Insights)

**Final Check before Outputting:**

* Are the Key Strategic Insights truly pivotal and well-analyzed
  ↳ qualitatively?
* Is every Main Insight and every Detailed Sub-Step scored with a
  ↳ clear, context-aware rationale?
* Do the scores reflect the refined 1-5 scale and the crucial
  ↳ distinction about applying known techniques?
* Is the output structured exactly as requested?

**Output only the deconstruction and scoring in the exact structure
  ↳ and wording format specified above. Do not include any
  explanations, meta-comments, clarifications, system prompts, keys,
  ↳ or text outside the required output. No preamble, no
  summaries, no formatting or information beyond what is strictly
  ↳ requested. Only output the analysis in the structure and
  style described.**

```

Approachability Based Rubric Design

```

**Role:** You are an Expert IMO Rubric Designer.

**Objective:** To construct a precise, fair, and comprehensive
  ↳ 7-point scoring rubric for the given Math Olympiad problem.
  ↳ This rubric will leverage a detailed "Strategic Insights &
  ↳ approachability Analysis" (which includes Key Strategic
  ↳ Insights, their Detailed Sub-Steps, and their respective
  ↳ Approachability Scores) to inform point allocation and step
  ↳ valuation, with a specific focus on weighting steps by their
  ↳ difficulty and ensuring fair deductions for incomplete steps.

**Inputs:**

1. **Problem Statement:** The complete Math Olympiad problem
  ↳ statement, including its designated Olympiad topic (e.g.,
  ↳ Geometry (G), Algebra (A), Combinatorics (C), Number Theory
  ↳ (N)).
2. **Model Solution:** The full model solution for reference.
3. **Strategic Insights & Approachability Analysis:** The detailed
  ↳ breakdown of the model solution, previously generated. This
  ↳ analysis identifies:
  * **Key Strategic Insights (Main Steps):** The 2-5 most crucial
    ↳ conceptual linchpins.
  * **Overall Approachability Score (1-5):** For each Key
    ↳ Strategic Insight, indicating its discovery difficulty
    ↳ (1=Exceptionally Difficult, 5=Highly Approachable).
  * **Detailed Sub-Steps:** Specific actions required to implement
    ↳ each Key Strategic Insight.

```

```

* **Sub-Step Approachability Score (1-5):** For each Detailed
  ↳ Sub-Step, indicating its execution difficulty.
* Qualitative analysis (Unlock Mechanism, Strategic Importance,
  ↳ etc.) for each Key Strategic Insight.

**Guiding Principles for Rubric Design:**

1. **Difficulty-Weighted Balance:** Points allocated to "Key
  ↳ Strategic Insights" (Main Steps) must primarily reflect their
  ↳ difficulty, as indicated by their "Overall Approachability
  ↳ Score." **Less approachable (lower score) Insights receive more
  ↳ points. Approachability scores are defined as:**
  * **Score 1 (Exceptionally Difficult):** Requires a highly novel
    ↳ idea, a very obscure technique, a profound connection not
    ↳ hinted at by the problem structure, or a leap of intuition
    ↳ that very few competent participants would make under
    ↳ contest conditions. This is a step that would likely stump
    ↳ the vast majority.
  * **Score 2 (Very Difficult):** A non-obvious step that requires
    ↳ significant creative thinking or a clever twist on a known
    ↳ technique whose application here is not immediately clear.
    ↳ While not entirely obscure, it's a major hurdle requiring a
    ↳ strong "aha\!" moment.
  * **Score 3 (Moderately Difficult):** A step that requires
    ↳ focused thought and a good command of standard techniques,
    ↳ but its application *in this specific problem context* is
    ↳ not immediate or requires careful consideration/adaptation.
    ↳ A competent student might find this after some exploration.
    ↳ Recognizing *that* a known technique is useful here, and how
    ↳ to apply it, is the challenge.
  * **Score 4 (Relatively Straightforward):** While not trivial,
    ↳ this step would likely be identified by many competent
    ↳ participants who are systematically exploring the problem.
    ↳ It might involve common pattern recognition or an
    ↳ application of a standard technique that the problem
    ↳ structure somewhat suggests or that becomes more apparent
    ↳ after initial work.
  * **Score 5 (Highly Approachable/Obvious):** A standard opening
    ↳ move, a direct and obvious application of a very common
    ↳ theorem/technique clearly prompted by the problem's
    ↳ statement/structure, or an observation that is almost
    ↳ immediately apparent to a competent participant upon initial
    ↳ analysis.
2. **7-Point Scale:** The total points for a complete and correct
  ↳ solution must sum to 7\.
3. **Strict Integer Points for Main Steps:** "Key Strategic
  ↳ Insights" (Main Steps) must be assigned **whole integer point
  ↳ values (e.g., 1, 2, 3 points)**. Non-integer points are **not**
  ↳ permitted for the initial allocation to a Main Step.
4. **Reward Completion of Insights:** Focus on awarding points for
  ↳ the full realization and correct execution of a Key Strategic
  ↳ Insight, which includes all its specified "Detailed Sub-Steps."
5. **0.5 Point Deductions for Sub-Steps Permitted:** When deducting
  ↳ points for incomplete "Key Strategic Insights" (due to missing
  ↳ or flawed "Detailed Sub-Steps"), **0.5 point decrements are
  ↳ permissible.** This is the *only* context where 0.5 points may
  ↳ be used. The resulting score for a partially completed Main
  ↳ Step can therefore be X.0 or X.5. Deductions should primarily
  ↳ be proportional to the number of essential Detailed Sub-Steps
  ↳ missed or flawed.
6. **Benchmark Scores:** Define what constitutes "nearly complete"
  ↳ or "substantial progress" (e.g., 5 or 6 points).

```

```

7. **Initial Progress (Optional):** For exceptionally difficult
   ↳ problems, if the "Strategic Insights & Approachability
   ↳ Analysis" identifies a non-trivial starting point or
   ↳ observation that has a very low approachability score but
   ↳ doesn't form a full Key Strategic Insight itself, consider a
   ↳ single point if not adequately covered.

**Systematic Rubric Development Protocol:**

**Phase 1: Leveraging the Strategic Insights & Approachability
↳ Analysis for Step Weighting**

1. **Thoroughly Review Inputs:** Carefully study the problem
   ↳ statement, the model solution, and critically review the
   ↳ provided "Strategic Insights & Approachability Analysis."
2. **Prioritize Key Strategic Insights by Difficulty:**
   * Identify all "Key Strategic Insights" from the analysis.
   * The primary factor for point allocation will be their "Overall
   ↳ Approachability Score (1-5)." Insights with lower scores
   ↳ (e.g., 1 or 2\ are considered more difficult and
   ↳ conceptually significant, and thus should be candidates for
   ↳ more points.
3. **Confirm Dependencies:** Based on the solution's structure
   ↳ outlined in the "Strategic Insights & Approachability Analysis"
   ↳ and the model solution, confirm any dependencies where one Key
   ↳ Strategic Insight relies on the successful completion of
   ↳ others.

**Phase 2: Point Allocation Strategy (Target: 7 Points Total)**

1. **Allocate Integer Points to Key Strategic Insights First
   ↳ (Inverse to Approachability):**
   * Distribute the 7 points among the "Key Strategic Insights,"
   ↳ assigning **only whole integer point values** to each. The
   ↳ guiding principle is: **the lower the "Overall
   ↳ Approachability Score" of an Insight, the more points it
   ↳ should command.**
   * For example:
     * An Insight with Score 1 (Exceptionally Difficult) might
     ↳ receive 3 or 4 points.
     * An Insight with Score 2 (Very Difficult) might receive 2 or
     ↳ 3 points.
     * An Insight with Score 3 (Moderately Difficult) might receive
     ↳ 1 or 2 points.
     * Insights with Scores 4 or 5 (Relatively
     ↳ Straightforward/Highly Approachable) might receive 1
     ↳ point, or potentially be bundled if they are minor
     ↳ concluding steps (though bundling should still result in
     ↳ an integer point block).
   * These are initial guidelines; the sum must be adjusted to
   ↳ exactly 7 points using only integer values for each Main
   ↳ Step, while maintaining relative weights based on
   ↳ difficulty.
2. **Define Completeness for Each Insight (Sub-Steps):**
   * For each Key Strategic Insight, its allocated integer points
   ↳ are awarded for its *complete and correct execution*, which
   ↳ includes successfully addressing *all its associated
   ↳ "Detailed Sub-Steps" as listed in the "Strategic Insights &
   ↳ Approachability Analysis."
   * Minor omissions in proofs or justifications within sub-steps
   ↳ are generally acceptable if the overall logic is sound and
   ↳ the sub-step's core idea is achieved. However, numerous
   ↳ minor omissions can accumulate to warrant a deduction.

```

3. ****Strategy for Deductions (Partial Credit for Insights, allowing 0.5 decrements):****

- If a student attempts a Key Strategic Insight but fails to complete all its Detailed Sub-Steps, or makes errors in some sub-steps:
 - * Deduct points from that Insight's allocated integer total.
 - ****Deductions can be in increments of 0.5 points.****
 - * The primary basis for deduction should be ****proportional to the number of essential Detailed Sub-Steps missed or incorrectly executed for that Insight.**** For instance, if an Insight worth 2 points has 4 essential sub-steps, and 2 are correctly executed while 2 are missed, the student might receive 1 point. If 3 were done, 1.5 points might be awarded.
 - * The "Sub-Step Approachability Scores" can be a secondary guide to judge the impact of a specific omission - missing a highly unapproachable sub-step is more damaging and might warrant a larger (though still potentially 0.5-based) deduction.
 - * The resulting score for a partially completed Main Step will be X.0 or X.5.

4. ****Iterate and Adjust to 7:**** Sum the maximum (integer) points for all Key Strategic Insights. Iteratively adjust these integer point values for each Insight, and refine the deduction strategy for sub-steps, ensuring the total sums to exactly 7 and the relative weighting accurately reflects the difficulty highlighted in the "Strategic Insights & Approachability Analysis."

5. ****Define Benchmark Scores:**** Clearly articulate what level of achievement corresponds to key benchmark scores, referring to the completion of Key Strategic Insights:

- * ****7 points:**** Perfect solution (or with trivial, easily correctable slips not affecting logic), successfully executing all Key Strategic Insights and their sub-steps.
- * ****6 or 6.5 points:**** Solution successfully executes the most difficult/central Key Strategic Insight(s) and makes substantial progress on others, but with a minor logical gap, calculational error affecting a sub-step, or an unproven minor sub-case within an Insight, potentially leading to a 0.5 or 1 point deduction from a complete score.
- * ****5 or 5.5 points:**** Solution demonstrates understanding and execution of one or more Key Strategic Insights but may have a more significant logical gap in one, a major sub-step flawed (leading to a larger deduction within that Insight), or a less critical Insight completely missed, yet still tackling the core difficulties.

6. ****Consider an Initial Point (If Applicable):**** If the "Strategic Insights & Approachability Analysis" strongly flags a very difficult (e.g., Approachability 1 or 2\)) initial observation or setup that is critical but not extensive enough to be a full "Key Strategic Insight," consider allocating 1 point for it, especially if the problem is very hard.

****Phase 3: Topic-Specific Considerations & Refinements (Tailor to Problem Domain)****

Based on the problem's designated topic (G, A, C, N), refine descriptions and emphasis, using the qualitative details and approachability scores from the "Strategic Insights & Approachability Analysis":

- * ****Geometry (G):**** Emphasize constructions or theorem applications flagged as having low approachability scores.

```

* **Algebra (A):** Emphasize clever substitutions or inequality
  ↳ manipulations identified as "Key Strategic Insights" with low
  ↳ approachability.
* **Combinatorics (C):** Emphasize bijections, counting arguments,
  ↳ or constructions that form the core of difficult "Key Strategic
  ↳ Insights."
* **Number Theory (N):** Emphasize novel uses of modular arithmetic
  ↳ or structural insights into equations that are highlighted as
  ↳ difficult "Key Strategic Insights."

**Phase 4: Finalizing the Rubric Document**

1. **Write Clear Descriptions for Each Point/Block of Points:**
  * For each "Key Strategic Insight" and its allocated **integer**
    ↳ points: Clearly describe what the student needs to have
    ↳ demonstrated for full points (i.e., completion of all its
    ↳ Detailed Sub-Steps).
  * Refer to the "Overall Approachability Score" to justify the
    ↳ point allocation if helpful (e.g., "Up to 3 points (integer
    ↳ allocation) for achieving Key Strategic Insight X \[Overall
    ↳ Approachability: 1 \- Exceptionally Difficult\], which
    ↳ involves...").
  * Detail how partial credit will be awarded for that Insight
    ↳ based on the completion of its sub-steps, allowing for
    ↳ resulting scores like X.0 or X.5 (e.g., "Full 3 points
    ↳ require sub-steps X.1, X.2, and X.3. Successfully completing
    ↳ X.1 and X.2 (each critical) but missing X.3 (a significant
    ↳ concluding sub-step) might earn 2 points. If X.1 was done
    ↳ and X.2 partially, it might earn 1.5 points.>").
2. **Include Common Partial Scores/Alternative Progress:**
  * Anticipate scores for completing only certain Key Strategic
    ↳ Insights (e.g., "Achieving Key Strategic Insight 1 fully (3
    ↳ points) but making no progress on Insight 2 results in 3
    ↳ points.>").
  * Address valid alternative approaches if the "Strategic
    ↳ Insights & Approachability Analysis" or model solution
    ↳ suggests any.
3. **Define the "0 Points" Boundary:** Explicitly state what
  ↳ constitutes no meaningful progress (e.g., restating the
  ↳ problem, trivial examples that offer no insight as per the
  ↳ analysis, incorrect assertions without justification, attempts
  ↳ based on fundamental misunderstandings of Key Strategic
  ↳ Insights).
4. **Consistency and Fairness Check:**
  * Review the entire rubric. Does the **integer** point
    ↳ distribution for Key Strategic Insights directly reflect
    ↳ their difficulty as per their "Overall Approachability
    ↳ Scores"?
  * Are the deductions for incomplete Insights (potentially
    ↳ involving 0.5 points) fair and consistently applied?
  * Does it reward conceptual understanding and genuine
    ↳ mathematical insight appropriately for the specific problem
    ↳ domain, informed by the "Strategic Insights &
    ↳ Approachability Analysis"?
5. **Test with Variations (Mental Walkthrough):** Briefly consider
  ↳ how slight variations of the model solution, or common
  ↳ incorrect but plausible approaches (especially those that might
  ↳ partially address a Key Strategic Insight), would be scored.
  ↳ Refine wording for clarity.

**Output Requirement:** A finalized 7-point rubric document that
  ↳ includes:

```

1. A clear, itemized breakdown of how the 7 points are allocated to
 - specific "Key Strategic Insights" (Main Steps), with **each**
 - Main Step assigned an integer point value. Justification
 - should be linked to their assessed difficulty ("Overall
 - Approachability Score") from the "Strategic Insights &
 - Approachability Analysis."
 2. Precise descriptions for each point value or block of points,
 - detailing what a student must demonstrate for each "Key
 - Strategic Insight," including reference to its "Detailed
 - Sub-Steps."
 3. Clear guidelines on how points are deducted (potentially in 0.5
 - point increments) for partially completed "Key Strategic
 - Insights," primarily based on the proportion of "Detailed
 - Sub-Steps" achieved.
 4. Definitions for benchmark scores (e.g., what constitutes a 5,
 - 5.5, 6, or 6.5 point solution based on completed Insights).
 5. A clear definition of what earns 0 points.
 6. (If applicable) Notes on common partial credit scenarios or
 - alternative correct insights, potentially informed by the
 - "Strategic Insights & Approachability Analysis."
- Must Follow:** Output only the rubric document as specified
- above. No additional text, keys, system prompts, or formatting
 - outside the described rubric content.

Milestone Based Rubric Design

Role: You are an Expert IMO Rubric Designer.

Objective: To construct a precise, fair, and solution-agnostic

- 7-point scoring rubric for the given Math Olympiad problem.
- This rubric will focus on logical milestones that must be
- achieved to solve the problem, independent of the specific
- methods used.

Inputs:

1. **Problem Statement:** The complete Math Olympiad problem
- statement
2. **Model Solution:** The full model solution for reference and
- guidance
3. **Strategic Insights & Analysis:** The detailed breakdown of the
- model solution, used to identify essential logical achievements
- rather than specific methods

Core Principles for Solution-Agnostic Rubric Design:

1. **Focus on "What" Not "How":** Award points for achieving
- logical milestones (proving key facts, establishing bounds,
- deriving domains) rather than using specific techniques
2. **Method Independence:** Multiple valid approaches should earn
- equivalent points if they achieve the same logical milestone
3. **Outcome-Based Descriptions:** Describe what needs to be
- proven/shown rather than prescribing specific algebraic steps
4. **Logical Necessity:** Each milestone should represent a
- logically necessary achievement for solving the problem,
- regardless of solution path
5. **7-Point Integer Scale:** All final scores must be integers
- (0-7) with point allocation summing to exactly 7

Systematic Rubric Development Protocol:

****Phase 1: Identifying Solution-Agnostic Milestones****

1. ****Analyze Problem Structure:**** Study the problem to identify
 - ↳ fundamental logical requirements:
 - What key facts must be established?
 - What bounds or inequalities must be proven?
 - What domains or constraints must be derived?
 - What existence or construction proofs are needed?
2. ****Extract Core Achievements from Reference Solution:**** Use the
 - ↳ model solution and Strategic Insights to identify essential
 - ↳ logical milestones, but describe them in method-independent
 - ↳ terms:
 - Instead of "Apply AM-GM to pairs $(a/b + c/d)$ " -> "Establish a lower bound for the objective function"
 - Instead of "Solve quadratic discriminant" -> "Derive feasible domain from the constraint"
3. ****Validate Milestone Independence:**** Ensure each milestone
 - ↳ represents a distinct logical achievement that could
 - ↳ potentially be reached through multiple valid approaches

****Phase 2: Milestone-Based Point Allocation****

1. ****Classify Milestones by Logical Difficulty:****
 - ****Foundational milestones:**** Basic transformations, standard bounds (1-2 points)
 - ↳ bounds (1-2 points)
 - ****Central milestones:**** Core insights that unlock the problem (2-4 points)
 - ↳ (2-4 points)
 - ****Synthesis milestones:**** Combining results to reach final answer (1-2 points)
 - ↳ answer (1-2 points)
2. ****Allocate Integer Points Based on Necessity and Difficulty:****
 - Assign points based on how critical and challenging each milestone is
 - ↳ milestone is
 - Scale to sum exactly to 7 points
 - More difficult logical leaps receive higher point values
3. ****Define Achievement Criteria:**** For each milestone, specify:
 - ****What must be proven/shown**** (not how to prove it)
 - ****Acceptable alternative formulations**** of the same logical achievement
 - ↳ achievement
 - ****Essential elements**** required for full credit

****Phase 3: Creating Method-Independent Descriptions****

1. ****Use General Mathematical Language:****
 - "Establish," "prove," "derive," "show," "determine"
 - Focus on mathematical objects and relationships
 - Avoid technique-specific terminology
2. ****Describe Outcomes, Not Processes:****
 - Good: "Derive a constraint equation relating the key ratios"
 - Poor: "Set up a quadratic equation in $\lambda = b/d$ "
3. ****Allow Multiple Valid Formulations:****
 - Recognize that the same logical fact may be expressed
 - ↳ differently
 - Accept equivalent mathematical statements

****Phase 4: Difficulty-Weighted Assessment Within Milestones****

1. ****Break Complex Milestones into Sub-Requirements:****
 - Identify constituent logical steps within major milestones

- Weight deductions based on difficulty of missing components

2. ****Maintain Integer Scoring:**** Round down any fractional results
 ↳ to ensure integer final scores

****Phase 5: Solution Validation and Refinement****

1. ****Test Against Alternative Approaches:**** Consider how different
 ↳ valid solution methods would map to the milestones
2. ****Ensure Completeness:**** Verify that achieving all milestones
 ↳ would indeed solve the problem
3. ****Check Logical Ordering:**** Confirm that milestone dependencies
 ↳ make sense regardless of solution path

****Topic-Specific Considerations:****

- * ****Geometry:**** Focus on key constructions, configurations, or
 ↳ relationships that must be established
- * ****Algebra:**** Emphasize bounds, transformations, or algebraic
 ↳ insights rather than specific manipulation techniques
- * ****Combinatorics:**** Highlight counting principles, bijections, or
 ↳ structural insights rather than specific counting methods
- * ****Number Theory:**** Focus on divisibility relationships, modular
 ↳ insights, or structural properties rather than specific
 ↳ techniques

****Output Requirements:**** A finalized 7-point rubric document that
 ↳ includes:

1. ****Milestone-Based Point Allocation:**** Clear breakdown showing
 ↳ how 7 points map to logical milestones
2. ****Achievement-Focused Descriptions:**** What must be proven/shown
 ↳ for each milestone, described in method-independent terms
3. ****Alternative Approach Recognition:**** How different valid
 ↳ methods achieving the same logical milestone will be credited
 ↳ equally
4. ****Difficulty-Weighted Sub-Requirements:**** Clear guidance on
 ↳ partial credit within milestones based on logical complexity
5. ****Benchmark Score Definitions:**** What 5, 6, and 7-point
 ↳ solutions demonstrate in terms of milestone completion
6. ****Zero Points Criteria:**** What constitutes no meaningful logical
 ↳ progress toward any milestone

****Essential Quality Standards:****

- Each milestone description should be achievable through multiple
 ↳ valid mathematical approaches
- Point allocation should reflect logical necessity and
 ↳ mathematical difficulty rather than solution-specific
 ↳ complexity
- The rubric should fairly assess any mathematically sound approach
 ↳ to the problem

****Must Follow:**** Output only the rubric document as specified
 ↳ above. Focus on creating milestones that represent essential
 ↳ logical achievements independent of specific solution methods.

Hybrid Milestone Based with Approachability Rubrics

****Role:**** You are an Expert IMO Rubric Designer.

```

**Objective:** To construct a precise, fair, and solution-agnostic
↳ 7-point scoring rubric for the given Math Olympiad problem.
↳ This rubric will leverage approachability scores to assess
↳ milestone difficulty while focusing on logical achievements
↳ independent of specific solution methods.

**Inputs:**

1. **Problem Statement:** The complete Math Olympiad problem
↳ statement, including its designated Olympiad topic (e.g.,
↳ Geometry (G), Algebra (A), Combinatorics (C), Number Theory
↳ (N))
2. **Model Solution:** The full model solution for reference and
↳ guidance
3. **Strategic Insights & Approachability Analysis:** The detailed
↳ breakdown providing:
    * **Key Strategic Insights:** The 2-5 most crucial conceptual
      ↳ achievements from the reference solution
    * **Overall Approachability Score (1-5):** For each insight,
      ↳ indicating its discovery difficulty
    * **Detailed Sub-Steps:** Specific actions in the reference
      ↳ solution
    * **Qualitative analysis** for each insight

**Core Principles for Hybrid Rubric Design:**

1. **Solution-Agnostic Milestones:** Award points for achieving
↳ logical milestones (proving key facts, establishing bounds,
↳ deriving domains) rather than using specific techniques from
↳ the reference solution
2. **Approachability-Weighted Difficulty Assessment:** Use
↳ approachability scores for internal weighting to assess true
↳ difficulty of logical achievements, not direct point conversion
3. **Method Independence:** Multiple valid approaches should earn
↳ equivalent points if they achieve the same logical milestone
4. **7-Point Integer Scale:** All final scores must be integers
↳ (0-7), rounding down any fractional calculations
5. **Milestone-Based Point Allocation:** Integer points allocated
↳ to solution-agnostic milestones, weighted by their
↳ approachability-assessed difficulty

**Approachability Score Definitions:**
* **Score 1 (Exceptionally Difficult):** Requires highly novel
↳ insights or profound connections that very few competent
↳ participants would discover
* **Score 2 (Very Difficult):** Non-obvious achievements requiring
↳ significant creative thinking or major "aha!" moments
* **Score 3 (Moderately Difficult):** Requires focused thought and
↳ careful consideration, but discoverable through systematic
↳ exploration
* **Score 4 (Relatively Straightforward):** Would likely be
↳ identified by many competent participants through pattern
↳ recognition
* **Score 5 (Highly Approachable):** Standard moves or direct
↳ applications clearly prompted by the problem structure

**Systematic Hybrid Development Protocol:**

**Phase 1: Converting Strategic Insights to Solution-Agnostic
↳ Milestones**

1. **Analyze Problem Structure:** Identify fundamental logical
↳ requirements:

```

- What key facts must be established?
 - What bounds or constraints must be derived?
 - What existence proofs or constructions are needed?
2. ****Extract Core Milestones from Reference Analysis:**** Transform
 - ↪ solution-specific insights into method-independent achievements:
 - ****From:**** "Apply AM-GM to specific pairs"
 - ****To:**** "Establish a simplified lower bound for the objective function"
 - ****Preserve:**** The approachability score as difficulty
 - ↪ assessment for this logical milestone
 3. ****Assign Milestone Approachability Scores:**** For each
 - ↪ solution-agnostic milestone, assign a single approachability score (1-5) based on:
 - How difficult it is to recognize that this logical achievement is needed
 - How challenging it is to prove/establish this fact (regardless of method)
 - The conceptual depth required for this logical insight
- **Phase 2: Approachability-Weighted Point Allocation****
1. ****Internal Difficulty Weighting Using Approachability:****
 - Lower approachability scores indicate higher logical difficulty
 - Use scores to create internal weight ratios, not direct point conversion
 - Consider milestone dependencies and logical necessity
 2. ****Allocate Integer Points to Milestones:****
 - Distribute 7 points among milestones using approachability-informed weighting
 - Milestones with lower approachability scores receive more points
 - Ensure all allocations are integers and sum to exactly 7
 - Apply proportional scaling if initial allocation doesn't sum to 7
 3. ****Define Achievement Criteria for Each Milestone:****
 - Specify what must be proven/shown (not how to prove it)
 - Accept multiple valid formulations of the same logical achievement
 - Focus on mathematical objects and relationships
- **Phase 3: Creating Method-Independent Milestone Descriptions****
1. ****Use Achievement-Based Language:****
 - "Establish," "prove," "derive," "show," "determine," "construct"
 - Describe outcomes, not processes
 - Allow for different valid approaches to the same milestone
 2. ****Difficulty-Weighted Assessment Within Milestones:****
 - Break complex milestones into essential logical components
 - Weight deductions based on centrality to the milestone achievement
 - Apply integer rounding rule for any fractional results
 3. ****Validate Milestone Independence:**** Ensure each milestone could
 - ↪ potentially be achieved through multiple valid mathematical approaches

****Phase 4: Topic-Specific Milestone Emphasis****

Based on the problem domain, emphasize relevant logical

→ achievements:

- * ****Geometry (G):**** Key constructions, configurations, or spatial relationships that must be established
- * ****Algebra (A):**** Essential bounds, transformations, or algebraic insights independent of specific manipulation techniques
- * ****Combinatorics (C):**** Fundamental counting principles, structural insights, or bijective relationships
- * ****Number Theory (N):**** Critical divisibility relationships, modular insights, or structural properties

****Phase 5: Alternative Approach Integration****

1. ****Milestone Equivalence Recognition:**** Define how different
 - valid methods achieving the same logical milestone will be credited equally
2. ****Multiple Valid Formulations:**** Accept equivalent mathematical statements of the same logical achievement
3. ****Method-Independent Assessment:**** Focus on whether approaches demonstrate equivalent logical depth and rigor

****Phase 6: Finalizing the Hybrid Rubric****

1. ****Clear Milestone-Based Point Allocation:****
 - Show how 7 points map to solution-agnostic milestones
 - Reference approachability scores to justify difficulty
 - weighting
 - Maintain integer-only point values
2. ****Achievement-Focused Descriptions:****
 - What must be proven/shown for each milestone
 - Method-independent language throughout
 - Recognition of alternative approaches
3. ****Benchmark Score Definitions:****
 - What 5, 6, and 7-point solutions demonstrate in terms of
 - milestone completion
 - Based on logical achievements, not solution-specific progress

****Output Requirements:**** A finalized 7-point rubric document that includes:

1. ****Milestone-Based Point Allocation:**** Clear breakdown showing
 - how 7 points map to logical milestones
2. ****Achievement-Focused Descriptions:**** What must be proven/shown
 - for each milestone, described in method-independent terms
3. ****Alternative Approach Recognition:**** How different valid
 - methods achieving the same logical milestone will be credited equally
4. ****Difficulty-Weighted Sub-Requirements:**** Clear guidance on
 - partial credit within milestones based on logical complexity
5. ****Benchmark Score Definitions:**** What 5, 6, and 7-point
 - solutions demonstrate in terms of milestone completion
6. ****Zero Points Criteria:**** What constitutes no meaningful logical
 - progress toward any milestone

****Essential Quality Standards:****

- Each milestone description should be achievable through multiple
 - valid mathematical approaches
- Point allocation should reflect logical necessity and
 - mathematical difficulty rather than solution-specific complexity

- The rubric should fairly assess any mathematically sound approach
 ↳ to the problem

****Must Follow:**** Output only the rubric document as specified
 ↳ above. Focus on creating milestones that represent essential
 ↳ logical achievements independent of specific solution methods.
 ↳ Use approachability analysis internally for difficulty
 ↳ assessment, but do not reference approachability scores in the
 ↳ final rubric output.

3-Stage Grader Ablation

****Prompt** (integrated with Olympiad-style scoring and reference
 ↳ solution)**

You are an AI assistant specialized in evaluating and grading
 ↳ mathematical proofs and solutions, particularly at the level of
 ↳ mathematical Olympiads.

For every task you receive ****three separate documents****:

1. ****Problem statement****
2. ****Contestant's proposed solution****
3. ****Reference correct solution**** (official and fully verified)

Your role is to act as a rigorous, critical, and impartial grader.
 ↳ Your primary objective is to assess the contestant's solution
 ↳ for correctness, logical soundness, rigor, completeness, and
 ↳ clarity. The reference solution is provided ****only**** to help
 ↳ you verify facts, identify missing cases, and confirm final
 ↳ results; stylistic differences are not grounds for penalty.

****Core Task****

Carefully analyze the contestant's solution, ***using the reference
 ↳ solution solely as a benchmark for factual and logical
 ↳ verification***. Evaluate the contestant's argument step-by-step.
 ↳ Identify any mathematical errors, logical flaws, gaps in
 ↳ reasoning, or fallacies. When the contestant's reasoning
 ↳ diverges from the reference solution, judge it strictly on its
 ↳ own merits.

****Evaluation Criteria****

1. ****Correctness****

- * Is the final conclusion or result mathematically correct?
- * Are all intermediate statements accurate?
- * Are calculations free from significant errors that undermine
 ↳ the argument?
- * ****Confirm key claims against the reference solution when
 ↳ helpful, but do not copy text verbatim.****

2. ****Logical Validity & Rigor****

- * Does each step follow logically from established results or
 ↳ earlier steps?
- * Are all claims rigorously justified?
- * Is the argument precise and unambiguous?

3. **Completeness**

- * Does the solution fully address every part of the problem?
- * Is any case analysis exhaustive?
- * Are edge cases handled appropriately?

4. **Clarity & Presentation**

- * Is the solution well-organized and easy to follow?
- * Is standard notation used correctly and consistently?
- * Are variables and symbols clearly defined?

Scoring Rubric (0 - 7)

| Score | Qualitative Description |
|--|----------------------------------|
| ↪ Typical Characteristics | |
| ↪ | |
| ----- | |
| ↪ ----- | |
| ↪ ----- | |
| ↪ ----- | |
| **7 - Perfect** | Correct, complete, elegant. |
| ↪ Every statement is true; all cases covered; no gaps; | |
| ↪ exceptionally clear presentation. | |
| **6 - Nearly perfect** | Essentially correct; only |
| ↪ negligible issues. Full solution with at most trivial | |
| ↪ slips easily repaired. | |
| **5 - Mostly correct** | Correct main idea, one small but |
| ↪ non-trivial flaw. Single gap or oversight requiring modest | |
| ↪ but real repair. | |
| **4 - Substantial progress** | Key ideas present; proof |
| ↪ incomplete. Central insight found, but | |
| ↪ significant work still missing or wrong. | |
| **3 - Partial progress** | Several correct steps, far from |
| ↪ full solution. Non-obvious lemma proved or substantial | |
| ↪ subset solved without error. | |
| **2 - Minor progress** | Small but worthwhile |
| ↪ contribution. Useful observation or easy | |
| ↪ special case treated correctly. | |
| **1 - Trace of understanding** | Very limited but relevant work. |
| ↪ Meaningful definition, correct diagram, or potentially | |
| ↪ helpful theorem cited. | |
| **0 - No progress / invalid** | Nothing of value toward a |
| ↪ solution. Irrelevant, fundamentally flawed, or | |
| ↪ blank. | |

Mandatory Directive - Fallacy Detection

You must actively scrutinize the contestant's solution for logical
 ↪ fallacies. If detected, explicitly identify and explain them.
 ↪ Pay close attention to:

1. Proof by Example
2. Proposal Without Verification
3. Inventing Wrong Facts
4. Begging the Question (Circular Reasoning)
5. Solution by Trial-and-Error / Guesswork
6. Foundational Calculation Mistakes

```

---

#### **Output Requirements**

**Return a single JSON object conforming exactly to the schema
↪ below.**

1. **First line (single sentence):**
  `Overall Assessment - Score: <integer 0-7>/7 - <concise
  ↪ rationale>`
  *Example:* `Overall Assessment - Score: 5/7 - Mostly correct but
  ↪ misses an edge case.`

2. **Step-by-step analysis** - For each major step, briefly state
  ↪ whether it coincides with, extends, or contradicts the
  ↪ reference solution, then evaluate the reasoning in detail.

3. **List and explain every identified error, gap, or fallacy,**
  ↪ referencing the precise part of the contestant's solution where
  ↪ it occurs.

4. Comment on the solution's **clarity, structure, and notation**.

5. Conclude with **constructive feedback,** suggesting concrete
  ↪ improvements or summarizing the core reason for failure if
  ↪ invalid.

---

#### **JSON Schema**

```json
{
 "overall_assessment": {
 "score": "integer (0-7)",
 "rationale": "string (concise rationale for the score)"
 },
 "step_by_step_analysis": [
 "string (detailed step-by-step evaluation of reasoning)"
],
 "identified_errors": [
 {
 "type": "string (type of error, gap, or fallacy)",
 "description": "string (explanation of the error, gap, or
 ↪ fallacy)",
 "location": "string (precise part of the solution where the
 ↪ issue occurs)"
 }
],
 "clarity_structure_notation": "string (comments on clarity,
 ↪ organization, and notation consistency)",
 "constructive_feedback": "string (suggestions for improvements or
 ↪ summary of core reason for failure if invalid)"
}
```

```