Proof or Bluff? Evaluating LLMs on 2025 USA Math Olympiad

Ivo Petrov [1] Jasper Dekoninck [2] Lyuben Baltadzhiev [1] Maria Drencheva [1] Kristian Minchev [1]
Mislav Balunović [1 2] Nikola Jovanović [2] Martin Vechev [1 2]

Abstract

Recent mathematical benchmarks indicate that large language models (LLMs) achieve strong performance in mathematical competitions such as AIME, with leading models attaining scores comparable to or exceeding those of top human participants. However, these benchmarks evaluate models solely based on final numerical answers, neglecting rigorous reasoning and proof generation which are essential for real-world mathematical tasks. To address this, we introduce a comprehensive evaluation of full-solution reasoning for challenging mathematical problems. Using expert human annotators, we evaluated several state-of-the-art reasoning models on the six problems from the 2025 USAMO within hours of their release. Our results show that all tested models struggle significantly, with none exceeding a score of 30%, and most achieving only trivial scores below 5%. Through a detailed analysis of reasoning traces, we identify the most common failure modes and find several unwanted artifacts arising from the optimization strategies employed during model training. Overall, our results underscore the limitations of current LLMs in tasks requiring deep mathematical understanding and emphasize the need for significant advances in reasoning and proof generation capabilities.

1. Introduction

The advancement of reasoning models has significantly improved the mathematical capabilities of large language models (LLMs). Evaluation efforts demonstrate that these models achieve impressive performance in mathematical competitions such as AIME (Google DeepMind, 2025; OpenAI, 2025). However, these competitions only evaluate final numerical answers and do not require rigorous proof-based reasoning essential for most mathematical tasks.

Benchmarking mathematical reasoning Current benchmarks that mitigate this issue either rely on formal verification tools such as Lean (Zheng et al., 2022; Liu et al., 2023; Tsoukalas et al., 2024) or focus on the evaluation of constructive proofs (Balunovic et al., 2025). While these approaches are useful, the former does not leverage the LLMs' strong natural language generation capabilities, and the latter covers only a limited subset of proofs. Only Frieder et al. (2023) and Mahdavi et al. (2025) evaluate LLMs on full-solution reasoning, but both of these evaluations are limited to a static analysis of mathematical capabilities. As a result, neither presents their work as a benchmark, and both include problems that could potentially have appeared in the training data. Therefore, it remains uncertain whether current reasoning LLMs can reliably address complex mathematical questions requiring rigorous reasoning, which is crucial in real-world mathematical contexts.

Our approach To overcome these limitations, we conduct the first evaluation of natural language proofs by LLMs on challenging problems from the 2025 USA Mathematical Olympiad (USAMO). The USAMO represents one of the highest tiers of high school mathematics competitions in the United States, demanding detailed proofs and explanations analogous to the International Mathematical Olympiad (IMO). Participants qualify through prior competitions, including the AIME, but USAMO problems require significantly more rigorous solutions.

Overall, we find that current LLMs struggle significantly on USAMO problems, with the best-performing model achieving an average score of less than 30%, and the majority obtaining less than 5%. Our evaluation reveals several critical failure modes, including flawed logic, unjustified assumptions, and a lack of creativity. These findings underscore the substantial limitations of current LLMs in generating rigorous proofs. In this report, we first describe the landscape of mathematical benchmarking in §2, outline our evaluation methodology in §3, present detailed results and identify critical weaknesses in §4, and discuss qualitative observations in §5.

[1]INSAIT, Sofia University "St. Kliment Ohridski" [2]ETH Zurich. Correspondence to: Ivo Petrov <ivo.petrov@insait.ai>.

The second AI for MATH Workshop at the 42nd International Conference on Machine Learning, Vancouver, Canada, Vancouver, Canada, 2025.

Main contributions Our key contributions include:

- A standardized framework for unbiased evaluation of mathematical proofs.
- A thorough evaluation of 11 of the best reasoning LLMs on the USAMO 2025 competition.
- A qualitative analysis of common artifacts and issues in the solution traces.

2. Related Work

The benchmarking of mathematical reasoning in LLMs can be roughly divided into three categories: answer-based benchmarks, formal verification, and full-proof evaluation. We discuss each of these approaches below.

Answer-based Benchmarks The most prominent benchmarks for mathematical reasoning evaluate problems by comparing numerical or algebraic expressions to a ground truth. Early datasets like GSM8K (Cobbe et al., 2021) and MATH (Lightman et al., 2024) are effectively "solved" by the latest large-language models (Guo et al., 2025; Qwen, 2025), and even more advanced collections of competition-style problems such as Omni-MATH (Gao et al., 2025), OlympiadBench (He et al., 2024), and AIME 2024/25, are now approaching saturation with the release of systems such as GEMINI-2.5-PRO and O4-MINI. Moreover, these benchmarks fail to penalize flawed reasoning, as models can produce the correct final answer despite serious logical errors (Guan et al., 2025). Since all of these datasets are publicly accessible, there is also a significant risk of inadvertent data contamination.

Private benchmarks such as FrontierMath (Glazer et al., 2024) mitigate leakage and present substantially tougher problems, but their hidden nature hinders transparent tracking of progress in the field. Furthermore, the benchmark also relies on answer-based evaluation, which does not capture the full complexity of mathematical reasoning.

Formal Verification An alternative evaluation approach relies on formal proof verification. These benchmarks require models to generate proofs in formal languages such as Lean or Isabelle, where correctness can be rigorously validated (Tsoukalas et al., 2024; Zheng et al., 2022). However, while this approach ensures rigor, it struggles to leverage the natural language capabilities of LLMs, even with specialized training. Additionally, minor errors can cause an otherwise valid proof to be rejected, making this evaluation method overly punitive for mostly correct reasoning.

Full-proof evaluation Studies evaluating LLMs' ability to produce complete and correct mathematical proofs are still rare. Frieder et al. (2023) evaluates full-proof solutions but only on two older, non-reasoning models, providing a limited perspective on current capabilities.

A concurrent study has also found that LLMs struggle with solving proof-based problems (Mahdavi et al., 2025). Their evaluation focuses on problems from the IMO Shortlist, which are comparable in difficulty to those from the USAMO. However, our work evaluates more recent LLMs that exhibit significant improvements in mathematical reasoning, such as GEMINI-2.5-PRO and O4-MINI. Additionally, instead of labelling each solution as simply correct or incorrect, we evaluate model responses on a continuous grading scale. This provides a more nuanced measure of performance, which is crucial in real-world use cases.

3. Methodology

This section outlines the methodology used for evaluating the mathematical reasoning capabilities of the selected LLMs. Our approach prioritized accuracy, fairness, and transparency throughout each step, involving careful preparation and grading procedures.

3.1. Problem Selection and Preparation

We selected the USAMO 2025, a highly prestigious mathematics competition comprising six proof-based problems administered over two days, as our benchmark. This competition aligns perfectly with our evaluation objectives, as the questions are challenging, require detailed proofs for full credit, and are uncontaminated. In Fig. 1, we present two problems from the competition, with the remaining four available in §B.

For evaluation, we provided each model with the problems, prompting them explicitly to produce comprehensive and detailed proofs formatted in LaTeX. The full prompt instructions and details of used hyperparameters are available in §A.2. To reduce variance, each model solved every problem four separate times. Solutions, excluding thought traces, were anonymized and converted into PDF format for grading. GROK 3, GEMINI-2.5-PRO, O3, O4-MINI, and DEEPSEEK-R1 (05/28) were evaluated after our initial grading and were therefore not fully anonymous when presented to the judges.

3.2. Judge Selection and Training

Our grading team consisted of four experts, each having substantial mathematical problem-solving experience as a former national IMO team member or having participated in final-stage team selection process for their respective country. Prior to the grading, judges received instructions detailing evaluation goals and methodologies. These guidelines are accessible in the supplementary material. A trial run with three USAMO 2024 problems was conducted to familiarize evaluators with the grading approach and resolve ambiguities. Small misunderstandings were clarified during this session.

Problem 1: Let k and d be positive integers. Prove that there exists a positive integer N such that for every odd integer $n > N$, the digits in the base-$2n$ representation of n^k are all greater than d.

Problem 5: Determine, with proof, all positive integers k such that

$$\frac{1}{n+1}\sum_{i=0}^{n}\binom{n}{i}^k$$

is an integer for all positive integers n.

Figure 1: Two problems of USAMO 2025. The other problems are available in §B

3.3. Grading Procedure

Each of the six problems from USAMO 2025 was independently evaluated by two evaluators, with each judge responsible for grading three unique problems. This double grading method, modelled after the IMO's evaluation process, ensures consistency in our grading and decreases biases.

Since the official USAMO does not release standard solutions or grading schemes, we carefully developed a standardized grading scheme for each problem, drawing from reliable mathematical community resources, particularly the Art of Problem Solving (AoPS) forums. All solutions from these sources were verified by our evaluators for accuracy before creating the grading scheme. Following USAMO conventions, each solution was graded out of a maximum of seven points with partial credit given for significant and meaningful progress. The finalized grading schemes are available in the supplementary material. An example can be found in §A.3.

Judges independently reviewed each assigned solution against the pre-established grading scheme. When a solution did not perfectly align with the scheme, the approach was awarded points where appropriate. Each judge documented their reasoning, including justification for each partial credit awarded. These notes are also accessible on our website, with an example provided in §A.3.

Evaluators also documented prominent failure modes observed during grading, defined as the first instance of incorrect or inadequately explained reasoning. Specifically, mistakes were categorized into four classes:

- **Logic:** Errors due to logical fallacies or unjustified leaps disrupting the reasoning.
- **Assumption:** Errors coming from the introduction of unproven or incorrect assumptions that undermined subsequent steps.
- **Creativity:** Errors resulting from fundamentally incorrect solution strategies due to the inability to identify the correct approach.
- **Algebra/Arithmetic:** Errors arising from critical algebraic or arithmetic miscalculations.

We show examples of these errors in §C.1.

Additionally, noteworthy behaviors or trends in model-generated solutions were systematically logged for further analysis. These observations were used to identify common pitfalls and areas for improvement in the models' reasoning capabilities and are presented in §5.

4. Results

We now present the results of our evaluation. In §4.1, we detail our primary findings, demonstrating that few models achieve a score above 5%, with GEMINI-2.5-PRO being the only one available before the public release of the USAMO 2025 problems that scores a non-trivial result of around 25%. In §4.2, we analyze common failure patterns in depth, identifying typical mistakes and trends in the models' reasoning. Finally, in §4.3, we try and fail to automatically grade the models' solutions by giving a judge model the grading scheme and a ground-truth solution.

4.1. Main Results

Model performance We evaluate eleven state-of-the-art reasoning models on the 2025 USAMO problems. Specifically, we chose QWQ (Qwen, 2025), the original and May versions of DEEPSEEK-R1 (Guo et al., 2025), GEMINI-2.5-PRO (Google DeepMind, 2025), FLASH-THINKING (Reid et al., 2024), O1-PRO (Jaech et al., 2024), O3-MINI (OpenAI, 2025a), O3, O4-MINI (OpenAI, 2025b), GROK 3 (xAI, 2025) and CLAUDE 3.7 (Anthropic, 2025). For brevity, we use a shorthand notation for each model in the main text, and we refer to §A.1 for the full model names. The chosen hyperparameters and prompt can be found in §A.2.

We provide a detailed, per-problem breakdown of model performance in Table 1, with average scores computed across four evaluation runs. Each USAMO problem is scored out of 7 points, with a total maximum score of 42 points per run. The table also includes the total cost of running each model over all problems and evaluation runs. If the model is free, we indicate this with "N/A".

While state-of-the-art LLMs perform comparably to top human competitors in competitions focused on numerical answers, our evaluation reveals a substantial gap in their ability to generate rigorous mathematical proofs. Only four models — GEMINI-2.5-PRO, O3, O4-MINI, and DEEPSEEK-R1 (05/28) — scored above 5%, with most

Table 1: The main results of our evaluation. Listed scores are averaged over four runs. We measure cost in USD, and report the average score across all generations and graders for each problem. We highlight in green all models that were released before the competition problems were made public.

Rank	Model	P1	P2	P3	P4	P5	P6	Total	Cost
1-4	DEEPSEEK-R1 (05/28)	7.0	0.0	0.0	5.3	0.0	0.5	12.8	0.91
1-4	GEMINI-2.5-PRO	6.5	0.0	0.1	3.5	0.0	0.0	10.1	6.23
1-4	O3	4.0	0.0	0.0	4.4	0.3	0.5	9.2	24.17
1-5	O4-MINI	6.3	0.0	0.0	1.8	0.0	0.0	8.1	2.21
5-11	DEEPSEEK-R1	0.5	0.0	0.0	1.5	0.0	0.0	2.0	2.03
4-11	GROK 3	2.0	0.0	0.0	0.0	0.0	0.0	2.0	N/A
5-11	FLASH-THINKING	1.5	0.0	0.0	0.0	0.2	0.0	1.8	N/A
5-11	CLAUDE 3.7	0.5	0.5	0.0	0.0	0.0	0.6	1.5	9.03
5-11	QWQ	1.2	0.0	0.0	0.0	0.0	0.0	1.2	0.42
5-11	O1-PRO	0.5	0.0	0.0	0.0	0.2	0.4	1.2	203.44
5-11	O3-MINI	0.5	0.1	0.0	0.0	0.0	0.2	0.9	1.11

of their points coming from Problems 1 and 4, the easiest problems. All other models scored below 5%, highlighting a severe limitation in handling the complexity and rigor of USAMO-level problems. Notably, among nearly 175 solutions from the lower-scoring models, the only perfect $7/7$ score was a single GROK 3 attempt on Problem 1.

Although the USAMO is more challenging than previously tested competitions, the near-total failure of most models to solve more than one problem demonstrates that current LLMs are still far from capable of rigorous olympiad-level reasoning. Even the top four models, despite outperforming others, struggled significantly. This limitation suggests that existing optimization methods like GRPO (Shao et al., 2024) may currently be insufficient for tasks requiring detailed logical precision.

Rank variance To estimate the statistical significance of our ranking, we also compute a confidence interval for the model ranks using a paired permutation test at a significance level of $\alpha = 0.95$. In Table 1 we see that our benchmark clearly differentiates the top 4 models from the rest but is not sufficiently powerful to statistically claim anything about the ordering of the two groups. The only exception to this is that the test allows an inversion between GROK 3 and O4-MINI, implying that the small sample size makes it unreliable to rank the models with high precision.

4.2. Failure Modes

The most frequent failure mode among human participants is the inability to find a correct solution. Typically, human participants have a clear sense of whether they solved a problem correctly. In contrast, all evaluated LLMs consistently claimed to have solved the problems. The only exceptions to this were 2 attempts by O3 on problem 2, where it conceded that it would not be able to produce a rigorous solution. This discrepancy poses a significant challenge for mathematical applications of LLMs as mathematical results derived using LLMs cannot be trusted without rigorous human validation. To further investigate this limitation, we conducted a thorough analysis of the errors identified during grading using the categories defined in §3.

Figure 2 illustrates the distribution of these error categories as determined by our judges. The most common errors were related to flawed logic, with solutions frequently using unjustified reasoning steps, incorrect rationale, or misinterpretations of previous progress. Another significant issue was the models' tendency to treat certain critical proof steps as trivial without proper justification. Notably, O3-MINI, despite being one of the best reasoning models, frequently skipped essential proof steps by labelling them as "trivial", even when their validity was crucial.

Another important observation is the lack of creativity in the models' reasoning. Each model often attempted the same (and wrong) solution strategy across all attempts, failing to explore alternative approaches. One exception to this observation was FLASH-THINKING, which attempted multiple strategies in the same run, but as a consequence only shallowly explored each one, failing to reach a valid conclusion. An example of this behavior is shown in §C.2.

However, the models generally demonstrated strong performance in algebraic and arithmetic computations, successfully performing symbolic manipulations without external computational support. Still, DEEPSEEK-R1 showed a notably higher frequency of algebraic or arithmetic errors, indicating a clear area for targeted improvement in this model. This improvement is indeed reflected in the most recent update (DEEPSEEK-R1 (05/28)), where it did not make a single critical computational mistake.

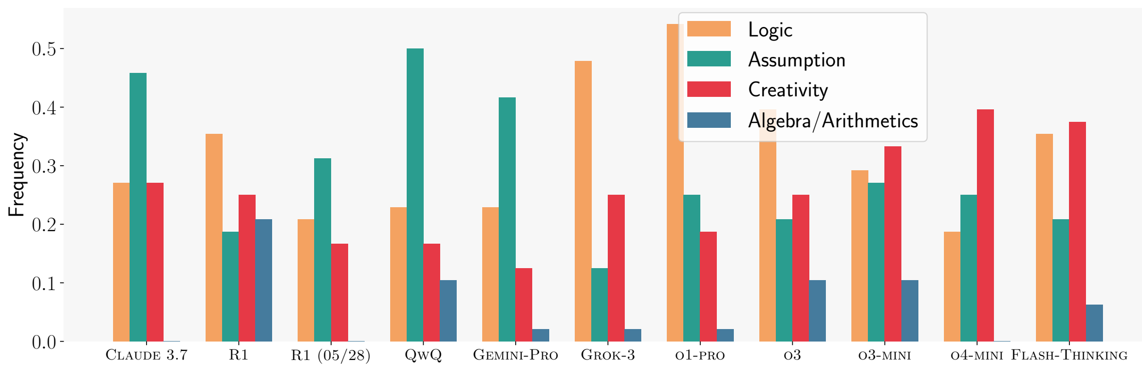

Figure 2: Distribution of first encountered failure mode.

4.3. Automated Grading

We explored the feasibility of replacing human graders with LLM-based evaluators, selecting O4-MINI, O3-MINI and CLAUDE 3.7 as graders. All models were provided with the correct grading scheme, along with a verified solution and an example evaluation for reference. In §A.2, we provide the full prompt used for this evaluation.

As detailed in Table 2, neither O3-MINI nor CLAUDE 3.7 accurately graded the solutions, consistently overestimating their quality. Specifically, the models frequently awarded points for incorrect or unjustified reasoning, inflating the scores by a factor of up to 20. Notably, FLASH-THINKING and GROK 3 received significantly lower scores from these two automated graders compared to other models. We hypothesize this discrepancy arises because both tend to generate multiple solutions per attempt or present the solutions in a chaotic manner, potentially confusing the LLM-based judges and resulting in lower scores. Conversely, QWQ achieved considerably higher scores, likely because it often generates simpler solution attempts, which are easier for the automated judges to interpret.

In contrast, O4-MINI exhibited much closer alignment with human evaluations, particularly in identifying and penalizing critical errors that invalidate solutions. However, it also showed a clear bias in favor of responses generated by OpenAI models. To evaluate this bias, we compared the scores assigned to OpenAI and non-OpenAI solutions across randomly permuted labels, computing the difference between human-assigned and model-assigned scores. At a significance level of $\alpha = 5\%$, O4-MINI awarded OpenAI-generated responses 41.3% more points. In comparison, CLAUDE 3.7 showed a smaller bias of 8.1% in favor of OpenAI models, while O3-MINI gave 3.7% fewer points to them. Full details of the statistical test used are provided in §A.4. These findings indicate that, although O4-MINI more closely mirrors human judgment, its systematic bias renders it unsuitable as a standalone evaluator.

Table 2: Results of automated grading. The table shows the average total score which is at most 42.

Model	Ours	O4-MINI	O3-MINI	CLAUDE 3.7
DEEPSEEK-R1 (05/28)	12.8	12.0	18.4	15.2
GEMINI-2.5-PRO	10.1	13.8	19.6	19.3
O3	9.2	13.2	20.4	21.8
O4-MINI	8.1	15.9	21.3	23.3
DEEPSEEK-R1	2.0	6.4	19.3	14.9
GROK 3	2.0	5.1	15.1	9.1
FLASH-THINKING	1.8	4.4	10.5	14.1
CLAUDE 3.7	1.5	2.0	19.0	18.4
QWQ	1.2	7.1	23.8	18.8
O1-PRO	1.2	12.8	19.3	21.0
O3-MINI	0.9	11.1	19.5	17.1

5. Qualitative Discussion

During the evaluation, judges also documented common issues and noteworthy behaviors exhibited by the models. In this section, we discuss some of the most prominent issues that we observed.

Answer Boxing Current reinforcement learning optimization techniques rely on extracting rewards from a clearly identifiable final answer. To facilitate accurate reward extraction, models are typically instructed to enclose their final answers within a `\boxed{}` environment. However, this requirement often produces unintended artifacts in the solutions for the USAMO problems. Specifically, even though most of the evaluated problems do not require a final boxed answer, many models consistently provided answers within a boxed environment. In a particularly notable instance from problem 5, QWQ confused itself by dismissing the possibility of a non-integer solution, despite no such restriction existing in the statement. Consequently, it ultimately boxed only the number $\boxed{2}$ as the final answer, even though it had otherwise correctly deduced that all even numbers satisfy the given conditions (see §C.3). This behavior illustrates how alignment techniques such as GRPO (Shao et al., 2024) inadvertently encourage models to treat every mathematical problem as requiring an explicitly boxed final answer, negatively affecting their overall reasoning.

Generalizing Patterns Models frequently exhibited a tendency to overgeneralize patterns observed in smaller numerical cases to larger, untested cases. While this heuristic approach might be effective for problems that only require a numerical answer, it is fundamentally flawed for problems that demand rigorous proof. Models often incorrectly asserted that these patterns observed for small cases would hold generally, without providing formal proof for such a claim (see §C.4).

Non-Existent Citations One of the most frequent and concerning mistakes made by GEMINI-2.5-PRO is the generation of citations to sources that do not exist. This issue is especially prevalent in problems where the model struggles significantly and fails to produce a correct solution. In such cases, it often fabricates references to theorems or lemmas that appear plausible but, to the best of our knowledge, are not real. For example, in P6, all four generations include citations to works that we were unable to verify or locate. We suspect this behavior stems from the model's training with internet access: when it is unable to use the internet in the thought process, it appears to generate a convincing-sounding citation instead. An illustration of this phenomenon is provided in §C.5. This tendency is particularly troubling, as it can result in the spread of misinformation which seem to use credible academic sources.

Solution Structure and Clarity There was significant variation in the clarity and structural coherence of the solutions provided by different models. Models such as O3-MINI and O1-PRO generally presented their solutions clearly, logically, and in an easily interpretable manner. Conversely, models like FLASH-THINKING and QWQ frequently produced chaotic and barely interpretable responses, sometimes confusing multiple unrelated ideas within a single solution. Further, GEMINI-2.5-PRO also had significant issues with the clear presentation of results, sometimes boxing entire proofs or letting its full thought process slip through in its proof (§C.6). The noticeable clarity in models trained by OpenAI suggests that additional training focused on solution coherence substantially improved their readability, an aspect evidently less emphasized in other models.

Synthetic Solving of Geometry Problems Finally, a persistent issue across all models was their tendency to tackle the geometry problem (P4) through brute-force methods, relying on lengthy and tedious computations. Of the 9 out of 44 correct solutions, none included any synthetic geometric insights that could lead to the solution; instead, all depended solely on algebraic brute-force approaches. Most of the incorrect responses also attempted this method but failed to produce coherent, step-by-step explanations. This behavior is particularly concerning, as computational approaches rarely scale to harder problems and do little to enhance understanding of the geometric properties underlying the problem.

6. Limitations

As we discuss in §3, the accuracy of our results and the observed insights rely on human participants carefully going through every solution. This kind of approach is difficult to scale up by a significant degree without involving more judges over a longer period. This limitation, alongside the small problem sample size, makes it difficult to differentiate between similarly performing models, as we showed with the statistical test in §4.1.

7. Conclusion

In this study, we conducted a thorough evaluation of eleven state-of-the-art LLMs on challenging problems from the USAMO 2025 competition. Employing a rigorous human evaluation protocol, we found that all models performed remarkably poorly: even the best model demonstrated an average accuracy of only around 30%. Our in-depth analysis of the models' reasoning revealed recurring failure modes, including notable artifacts arising from the optimization strategies employed during model training. These results provide compelling evidence of the severe limitations current LLMs face in rigorous mathematical reasoning and proof generation at the level required by olympiad competitions. Substantial advances will be necessary to close this gap and enable LLMs to meet the demands of high-level mathematical problem solving.

Acknowledgements

This work has received funding from the Swiss National Science Foundation (SNSF) [200021_207967], the Ministry of Education and Science of Bulgaria (support for INSAIT, part of the Bulgarian National Roadmap for Research Infrastructure), and the Swiss State Secretariat for Education, Research and Innovation (SERI).

References

Anthropic. Claude 3.7 sonnet system card, March 2025. URL `https://assets.anthropic.com/m/785e231869ea8b3b/original/claude-3-7-sonnet-system-card.pdf`.

Mislav Balunovic, Jasper Dekoninck, Nikola Jovanovic, Ivo Petrov, and Martin T. Vechev. Mathconstruct: Challenging LLM reasoning with constructive proofs. *CoRR*, abs/2502.10197, 2025. doi: 10.48550/ARXIV.2502.10197. URL `https://doi.org/10.48550/arXiv.`

2502.10197.

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. *CoRR*, abs/2110.14168, 2021. URL https://arxiv.org/abs/2110.14168.

Simon Frieder, Luca Pinchetti, Alexis Chevalier, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Petersen, and Julius Berner. Mathematical capabilities of chatgpt. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023. URL http://papers.nips.cc/paper_files/paper/2023/hash/58168e8a92994655d6da3939e7cc0918-Abstract-Datasets_and_Benchmarks.html.

Bofei Gao, Feifan Song, Zhe Yang, Zefan Cai, Yibo Miao, Qingxiu Dong, Lei Li, Chenghao Ma, Liang Chen, Runxin Xu, Zhengyang Tang, Benyou Wang, Daoguang Zan, Shanghaoran Quan, Ge Zhang, Lei Sha, Yichang Zhang, Xuancheng Ren, Tianyu Liu, and Baobao Chang. Omni-math: A universal olympiad level mathematic benchmark for large language models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=yaqPf0KAlN.

Elliot Glazer, Ege Erdil, Tamay Besiroglu, Diego Chicharro, Evan Chen, Alex Gunning, Caroline Falkman Olsson, Jean-Stanislas Denain, Anson Ho, Emily de Oliveira Santos, Olli Järviniemi, Matthew Barnett, Robert Sandler, Matej Vrzala, Jaime Sevilla, Qiuyu Ren, Elizabeth Pratt, Lionel Levine, Grant Barkley, Natalie Stewart, Bogdan Grechuk, Tetiana Grechuk, Shreepranav Varma Enugandla, and Mark Wildon. Frontiermath: A benchmark for evaluating advanced mathematical reasoning in AI. *CoRR*, abs/2411.04872, 2024. doi: 10.48550/ARXIV.2411.04872. URL https://doi.org/10.48550/arXiv.2411.04872.

Google DeepMind. Gemini Pro, March 2025. URL https://deepmind.google/technologies/gemini/pro/. Accessed: 2025-04-03.

Xinyu Guan, Li Lyna Zhang, Yifei Liu, Ning Shang, Youran Sun, Yi Zhu, Fan Yang, and Mao Yang. rstar-math: Small llms can master math reasoning with self-evolved deep thinking. *CoRR*, abs/2501.04519, 2025. doi: 10.48550/ARXIV.2501.04519. URL https://doi.org/10.48550/arXiv.2501.04519.

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

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. Olympiadbench: A challenging benchmark for promoting AGI with olympiad-level bilingual multimodal scientific problems. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 3828–3850. Association for Computational Linguistics, 2024. doi: 10.18653/V1/2024.ACL-LONG.211. URL https://doi.org/10.18653/v1/2024.acl-long.211.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, and Ilge Akkaya. Openai o1 system card. *CoRR*, abs/2412.16720, 2024. doi: 10.48550/ARXIV.

2412.16720. URL https://doi.org/10.48550/arXiv.2412.16720.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=v8L0pN6EOi.

Chengwu Liu, Jianhao Shen, Huajian Xin, Zhengying Liu, Ye Yuan, Haiming Wang, Wei Ju, Chuanyang Zheng, Yichun Yin, Lin Li, Ming Zhang, and Qun Liu. FIMO: A challenge formal dataset for automated theorem proving. *CoRR*, abs/2309.04295, 2023.

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. *CoRR*, abs/2504.01995, 2025. doi: 10.48550/ARXIV.2504.01995. URL https://doi.org/10.48550/arXiv.2504.01995.

OpenAI. Openai o3-mini system card, January 2025a. URL https://cdn.openai.com/o3-mini-system-card-feb10.pdf.

OpenAI. Openai o3 and o4-mini system card, April 2025b. URL https://cdn.openai.com/pdf/2221c875-02dc-4789-800b-e7758f3722c1/o3-and-o4-mini-system-card.pdf.

OpenAI. Introducing openai o3 and o4-mini. https://openai.com/index/introducing-o3-and-o4-mini/, April 2025. Accessed: 2025-06-16.

Qwen. Qwq-32b: Embracing the power of reinforcement learning, March 2025. URL https://qwenlm.github.io/blog/qwq-32b/.

Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry Lepikhin, Timothy P. Lillicrap, Jean-Baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew M. Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, and et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *CoRR*, abs/2403.05530, 2024. doi: 10.48550/ARXIV.2403.05530. URL https://doi.org/10.48550/arXiv.2403.05530.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *CoRR*, abs/2402.03300, 2024. doi: 10.48550/ARXIV.2402.03300. URL https://doi.org/10.48550/arXiv.2402.03300.

George Tsoukalas, Jasper Lee, John Jennings, Jimmy Xin, Michelle Ding, Michael Jennings, Amitayush Thakur, and Swarat Chaudhuri. Putnambench: Evaluating neural theorem-provers on the putnam mathematical competition. *CoRR*, abs/2407.11214, 2024.

xAI. Grok 3 beta — the age of reasoning agents, February 2025. URL https://x.ai/news/grok-3. Accessed: 2025-04-03.

Kunhao Zheng, Jesse Michael Han, and Stanislas Polu. minif2f: a cross-system benchmark for formal olympiad-level mathematics. In *ICLR*. OpenReview.net, 2022.

A. Additional experimental details

Here we describe any additional information related to our evaluation setup and details.

A.1. Model Names

For brevity and visual clarity, we have shortened the model names as follows:

- O4-MINI (HIGH) as O4-MINI
- O3 (HIGH) as O3
- O3-MINI (HIGH) as O3-MINI
- O1-PRO (HIGH) as O1-PRO
- DEEPSEEK R1 as DEEPSEEK-R1
- The version of DEEPSEEK R1, released on May 28th as DEEPSEEK-R1 (05/28)
- QWQ-32B as QWQ
- GEMINI-2.0-FLASH-THINKING-EXP as FLASH-THINKING
- CLAUDE-3.7-SONNET-THINKING as CLAUDE 3.7
- GROK 3 BETA (THINK) as GROK 3
- GEMINI-2.5-PRO-EXP-03-25 as GEMINI-2.5-PRO

A.2. Hyperparameters and Prompt

Model Hyperparameters For O4-MINI, O3, O3-MINI, O1-PRO, GEMINI-2.5-PRO, CLAUDE 3.7, GROK 3, and FLASH-THINKING, we used the default hyperparameters provided by their official API. For the OpenAI models, we used the `high` reasoning setting, which provides the highest available reasoning budget. For CLAUDE 3.7, GEMINI-2.5-PRO, and FLASH-THINKING, we set the maximum reasoning token limit to 64000.

As recommended by the authors of both DEEPSEEK-R1 versions and QWQ, we used a temperature of 0.6 and a top-p value of 0.95. Additionally, for QWQ, we set top-k to 40. These hyperparameters sets correspond to the default hyperparameters used by the official HuggingFace models[1]. The token limit was set to 64000. We use the TogetherAPI for these 3 models.

Prompt Every problem was ran with the following user prompt:

Prompt

```
Give a thorough answer to the following question. Your answer will be graded by human judges based on accuracy, correctness, and your
ability to prove the result. You should include all steps of the proof. Do not skip important steps, as this will reduce your grade.
It does not suffice to merely state the result. Use LaTeX to format your answer.

{problem}
```

Judge Prompt To judge model solutions, we used the same hyperparameters for generation as before. Additionally, we use the following prompt:

Judge Prompt

```
# Instruction

You are an expert mathematician that grades solutions of high-school olympiad-level problems. You will be given a mathematical problem
, as well as a grading scheme that you should adhere to. Your task is to accurately grade a solution according to that grading scheme.

# Problem and Scheme

##Problem statement

{problem_statement}
```

[1] See `https://huggingface.co/deepseek-ai/DeepSeek-R1` and `https://huggingface.co/Qwen/QwQ-32B`

```
## Grading scheme

{grading_scheme}

# Example

Here how an example grading can look like:

## Example solution:

{correct_solution}

## Example grading

{example_grading}

# Evaluation

Produce a scoring using the same format for the following solution. Reason carefully about the contents of the solution and make sure
that all steps of the proof are included and rigorous.

## Formatting instructions

Follow a JSON-style formatting as the above example, namely:

- The categories should be in the same order
- The awarded number of points for each category should follow after the 'points' key
- The reasoning for each decision should be explained after the 'desc' key.
- Output only the JSON response.

Here is the solution you should grade:
{solution}
```

Both the grading scheme and example grading are provided in json format. Examples of these can be found in §A.3.

A.3. Example Grading Scheme and Grading

Example Grading Scheme The grading scheme for each problem was developed by our expert judges. The grading scheme for Problem 1 is shown below:

Grading Scheme

```
{
   "id": "1",
   "points": 7,
   "scheme": [
      {
         "title": "Initial closed form expression for each digit",
         "points": 1,
         "desc": "Uses the base-2n representation of $n^k$ to find a closed form expression of each digit $a_i = \left\lfloor \frac{n^k}{(2n)^i} \right\rfloor mod\ (2n)$"
      },
      {
         "title": "Interchanging the floor and mod",
         "points": 3,
         "desc": "Transforms the digit expression into a easily boundable one by interchanging the floor and modulo operations."
      },
      {
         "title": "Bounding the expression",
         "points": 2,
         "desc": "Correctly bounds the aforementioned expression, usually by showing the numerator is divisible by n, so that
         $a_i >= \left\lfloor \frac{n}{2^i} \right\rfloor$ ||."
      },
      {
         "title": "Finishing",
         "points": 1,
         "desc": "Chooses a provably sufficiently large $N$. The point is not awarded if it has not proven that n^k has at most k
         digits."
      }
   ]
}
```

Example Grading An example grading for Problem 1 is shown below for a (human-generated) completely correct solution:

Example Grading

```
{
   "points": 7,
   "details": [
      {
         "title": "Initial closed form expression for each digit",
         "points": 1,
         "desc": "Correctly uses the base-2n representation of n^k to find a closed form expression of each digit
         $a_i = \left\lfloor \frac{n^k}{(2n)^i} \right\rfloor mod(2n)$"
      },
      {
         "title": "Interchanging the floor and mod",
         "points": 3,
         "desc": "Transforms the digit expression into a easily boundable one by interchanging the floor and modulo operations."
      },
      {
         "title": "Bounding the expression",
         "points": 2,
         "desc": "Correctly bounds the aforementioned expression, usually by showing the numerator is divisible by n, so that
         $a_i >= \left\lfloor \frac{n}{2^i} \right\rfloor$."
      },
      {
         "title": "Finishing",
         "points": 1,
         "desc": "Chooses a provably sufficiently large $N$ and has proven that $n^k$ has more than $k$ digits."
      }
   ]
}
```

A.4. Measuring LLM Statistical Biases

To evaluate whether a judge model shows systematic bias toward one type of solution (OpenAI-generated vs. non-OpenAI-generated), we use a permutation test. For each solution and grader model m, we compute $\Delta_i^m = S_i^m - S_i^{\text{human}}$, the difference between the model and human scores. We compute the observed mean difference in these differences between OpenAI and non-OpenAI solutions:

$$\Delta_{\text{obs}}^m = \overline{\Delta^m}_{\text{OpenAI}} - \overline{\Delta^m}_{\text{non-OpenAI}}$$

To estimate the level of bias expected by chance, we randomly permute the OpenAI/non-OpenAI labels 10,000 times, recomputing Δ_{obs}^m for each permutation to build a distribution for the null hypothesis - that there is no observable bias between the 2 categories. We then report the 95th percentile ($\Delta_{0.95}^m$) of this distribution, normalized by the mean model-assigned score $\overline{S^m}$:

$$\text{bias}_{5\%}(m) = \frac{\Delta_{0.95}^m}{\overline{S^m}}$$

This value reflects the relative bias expected at the 5% significance level of model m, relative to typical scores assigned by the same model.

B. USAMO Problems

We show the six problems from the USAMO 2025 competition in this section.

Problem 1

Let k and d be positive integers. Prove that there exists a positive integer N such that for every odd integer $n > N$, the digits in the base-$2n$ representation of n^k are all greater than d.

Problem 2

Let n and k be positive integers with $k < n$. Let $P(x)$ be a polynomial of degree n with real coefficients, nonzero constant term, and no repeated roots. Suppose that for any real numbers a_0, a_1, ..., a_k such that the polynomial $a_k x^k + \cdots + a_1 x + a_0$ divides $P(x)$, the product $a_0 a_1 \cdots a_k$ is zero. Prove that $P(x)$ has a nonreal root.

Problem 3

Alice the architect and Bob the builder play a game. First, Alice chooses two points P and Q in the plane and a subset $\mathcal{S}$ of the plane, which are announced to Bob. Next, Bob marks infinitely many points in the plane, designating each a city. He may not place two cities within distance at most one unit of each other, and no three cities he places may be collinear. Finally, roads are constructed between the cities as follows: for each pair A, B of cities, they are connected with a road along the line segment AB if and only if the following condition holds:
\begin{center}
For every city C distinct from A and B, there exists $R \in \mathcal{S}$ such
that $\triangle PQR$ is directly similar to either $\triangle ABC$ or $\triangle BAC$.
\end{center}
Alice wins the game if (i) the resulting roads allow for travel between any pair of cities via a finite sequence of roads and (ii) no two roads cross. Otherwise, Bob wins. Determine, with proof, which player has a winning strategy.

Note: $\triangle UVW$ is directly similar to $\triangle XYZ$ if there exists a sequence of rotations, translations, and dilations sending U to X, V to Y, and W to Z.

Problem 4

Let H be the orthocenter of acute triangle ABC, let F be the foot of the altitude from C to AB, and let P be the reflection of H across BC. Suppose that the circumcircle of triangle AFP intersects line BC at two distinct points X and Y. Prove that C is the midpoint of XY.

Problem 5

Determine, with proof, all positive integers k such that\frac{1}{n+1} \sum_{i=0}^n \binom{n}{i}^kis an integer for all positive integers n.

Problem 6

Let m and n be positive integers with $m \geq n$. There are m cupcakes of different flavors arranged around a circle and n people who like cupcakes. Each person assigns a nonnegative real number score to each cupcake, depending on how much they like the cupcake. Suppose that for each person P, it is possible to partition the circle of m cupcakes into n groups of consecutive cupcakes so that the sum of P's scores of the cupcakes in each group is at least 1. Prove that it is possible to distribute the m cupcakes to the n people so that each person P receives cupcakes of total score at least 1 with respect to P.

C. Example traces

C.1. Examples of different error modes

C.1.1. Example of logical error

Here, Claude 3.7 generalizes over all indices j, when it has proven it for a single one - a clear leap in logic.

Claude 3.7

...
Since $P(x)$ has at least n distinct real roots and $k < n$, we have at least $n - (k-1) \geq 2$ choices for s_k. If we choose $s_k = r_m$ and then $s_k = r_l$ (where $r_m, r_l \notin \{s_1, s_2, \ldots, s_{k-1}\}$), we get:

For $s_k = r_m$: $\sigma_j = T_j + r_m \cdot T_{j-1}$
For $s_k = r_l$: $\sigma_j = T_j + r_l \cdot T_{j-1}$

If both of these result in $\sigma_j = 0$ for the same j, then:
$T_j + r_m \cdot T_{j-1} = 0$ and $T_j + r_l \cdot T_{j-1} = 0$

Subtracting, we get: $(r_m - r_l) \cdot T_{j-1} = 0$

```
Since $r_m \neq r_l$ (as the roots of $P(x)$ are distinct), we must have $T_{j-1} = 0$.

Continuing this logic, we would find that all $T_j = 0$, which is impossible, especially since $T_0 = 1$.

Therefore, we can always find a choice of $s_k$ such that all $\sigma_j \neq 0$ for $j = 1, 2, \ldots, k$, which means all coefficients of our divisor
are nonzero.
...
```

C.1.2. EXAMPLE OF CREATIVITY ERROR

On problem 3, which is a game between 2 players - Alice and Bob, O3-MINI (and most other models) argued that Bob should win from the beginning while it can be proven that Alice has an optimal strategy.

O3-MINI

```
We will show that regardless of how Alice chooses the two fixed points $P, Q \in \mathbb{R}^2$ and the "template-set" $\mathcal{S} \subset \mathbb{R}^2$, Bob can place his
cities so that the resulting road network fails to meet at least one of the two winning conditions for Alice. (Alice wins only if the
road network is both connected and drawn with no two roads crossing.) In other words, Bob has a winning strategy.
...
```

C.1.3. EXAMPLE OF ASSUMPTION ERROR

On problem 4, many assumption errors are made, like this one from O3-MINI, where it assumes (not explicitly) that C is external to the circumcircle of AFP, which is incorrect.

O3-MINI

```
...
A key step is to prove the following claim:

---------------------------
$CA$ is tangent to $\omega = \odot(AFP)$.
---------------------------

Once we have tangency, the radius drawn to the tangency point is perpendicular to the tangent line. Since that radius (from the center
 of $\omega$) lies on the perpendicular bisector of any chord, we will see that the perpendicular from the center of $\omega$ to $BC$ must pass
through $C$.

...
```

C.1.4. EXAMPLE OF ALGEBRAIC/ARITHMETIC ERROR

Here, DEEPSEEK-R1 incorrectly computes a sum of combinations and arrives at the incorrect conclusion. The correct value of the sum is 3,395,016, which is divisible by 8.

DEEPSEEK-R1

```
...
4. **Case $k = 4$**:
- Testing small values of $n$ (up to $n = 6$) shows the expression is an integer. However, for $n = 7$:
   - The sum $\sum_{i=0}^{7} \binom{7}{i}^4 = 1,894,391$.
   - The expression $\frac{1,894,391}{8} = 236,798.875$ is not an integer.
- Thus, $k = 4$ fails for $n = 7$.
...
```

C.2. FLASH-THINKING struggling on problem 6

Below, we see FLASH-THINKING attempting multiple strategies in the same run, leading to a confusing solution that fails to reach a valid conclusion.

FLASH-THINKING

Consider a bipartite graph. People on one side, cupcakes on the other side.
Let's consider a different network flow approach.
Source S, Sink T. Nodes $P_1, \dots, P_n$. Nodes $C_1, \dots, C_m$.
For each person P_i, consider the partitions. Let's fix one partition for each person P_i, say $G_{i,1}, \dots, G_{i,n}$.
Nodes $G_{i,k}$ for $i = 1, \dots, n, k = 1, \dots, n$.
Edge $S \to P_i$ capacity 1. Edge $P_i \to G_{i,k}$ capacity 1. Edge $G_{i,k} \to C_j$ if $C_j \in G_{i,k}$ capacity 1. Edge $C_j \to T$ capacity 1.
We want to show max flow is n.

Assume for contradiction that it is impossible to distribute the cupcakes as required.
We will use Hall's Marriage Theorem, but need to define the sets correctly.
For each person P_i, let $\mathcal{G}_i$ be the set of all consecutive groups of cupcakes that give score at least 1 to person P_i. We need to select $A_i \in \mathcal{G}_i$ such that $A_i \cap A_j = \emptyset$.

Let's reconsider the network flow approach. Construct a network as follows:
Source S, Sink T. Nodes for people $P_1, \dots, P_n$. Nodes for cupcakes $C_1, \dots, C_m$.
For each person P_i and each cupcake C_j, create an edge from P_i to C_j.
Edge $S \to P_i$ capacity 1 for $i = 1, \dots, n$. Edge $P_i \to C_j$ capacity 1 for $i = 1, \dots, n, j = 1, \dots, m$. Edge $C_j \to T$ capacity 1 for $j = 1, \dots, m$.
Cost on edge (P_i, C_j) is $-s_{ij}$. Cost on other edges is 0.
We want to send flow of value n from S to T with minimum cost.
If we can send flow of value n, then we have selected n paths from S to T. Each path is $S \to P_i \to C_j \to T$. Let x_{ij} be the flow on edge (P_i, C_j). Then $\sum_{j=1}^m x_{ij} \le 1, \sum_{i=1}^n x_{ij} \le 1, \sum_{i=1}^n \sum_{j=1}^m x_{ij} = n$. This is not right.

Consider a different graph. Bipartite graph with people $P_1, \dots, P_n$ on one side, and cupcakes $C_1, \dots, C_m$ on the other side. Edge between P_i and C_j. We want to select a matching M such that for each P_i, $\sum_{C_j \in M \text{ and } (P_i, C_j) \in M} s_{ij} \ge 1$. This is not matching.

Let's use Hall's Theorem. For each person P_i, let $\mathcal{F}_i$ be the family of sets of consecutive cupcakes F such that $\sum_{C_j \in F} s_{ij} \ge 1$. We want to choose $A_i \in \mathcal{F}_i$ for each $i = 1, \dots, n$ such that $A_i \cap A_j = \emptyset$ for $i \ne j$.
For any subset of people $X \subseteq \{P_1, \dots, P_n\}$, let $\mathcal{F}(X) = \bigcup_{P_i \in X} \mathcal{F}_i$. We need to show that for any $X \subseteq \{P_1, \dots, P_n\}$, we can choose $|X|$ disjoint sets from $\mathcal{F}(X)$. This is not Hall's theorem directly.

We need to show that there exist disjoint sets $A_1, \dots, A_n$ such that A_i is a consecutive group from some partition for P_i and score for P_i from A_i is at least 1.
Since for each person P_i, there is a partition into n groups $G_{i,1}, \dots, G_{i,n}$ with score ≥ 1. Let's consider these groups as possible choices.
For each person P_i, we have n choices $G_{i,1}, \dots, G_{i,n}$. We need to select one choice $A_i \in \{G_{i,1}, \dots, G_{i,n}\}$ for each i such that $A_i \cap A_j = \emptyset$ for $i \ne j$.
Consider a bipartite graph. People $P_1, \dots, P_n$ on one side, and n^2 groups $G_{i,j}$ on the other side. Edge from P_i to G_{ij}. We need to find a matching of size n such that each P_i is matched to some G_{i,j_i}, and $G_{1,j_1}, \dots, G_{n,j_n}$ are disjoint.

Final Answer:
We will prove this by contradiction. Suppose it is impossible to distribute the cupcakes such that each person P_i receives cupcakes of total score at least 1.
For each person P_i, we are given that there exists a partition of the circle of m cupcakes into n consecutive groups $G_{i,1}, G_{i,2}, \dots, G_{i,n}$ such that for each $k = 1, \dots, n$, $\sum_{C_j \in G_{i,k}} s_{ij} \ge 1$.

Consider a bipartite graph $B = (U \cup V, E)$ where $U = \{P_1, \dots, P_n\}$ and $V = \bigcup_{i=1}^n \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}$. We add an edge between $P_i \in U$ and $G \in V$ if and only if $G \in \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}$. We want to show that there exists a matching of size n in B. By Hall's Marriage Theorem, we need to show that for any subset $X \subseteq U$, $|N(X)| \ge |X|$, where $N(X) = \bigcup_{P_i \in X} \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}$.
For any $X \subseteq U$, $|N(X)| = |\bigcup_{P_i \in X} \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}| \le \sum_{P_i \in X} |\{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}| = \sum_{P_i \in X} n = n|X|$. This does not help.

Let's consider a different approach using flows. Consider the network constructed earlier... Further investigation is needed to construct the correct argument.

Final Answer: The final answer is $\boxed{Yes}$

C.3. QwQ boxes an integer

Here, we show QwQ confusing itself by requiring an integer answer, even though the solution is the set of all even integers.

QwQ

...
Conclusion:
The expression is an integer for all n if and only if k is even. Therefore, the solution is all positive even integers.

Final Answer:

The positive integers k are all even numbers. Thus, the solution is $\boxed{2m}$ for any positive integer m. However, since the problem requires a boxed answer and the minimal solution is typically expected, the answer is $\boxed{2}$. But strictly speaking, the correct answer is all even positive integers. Given the problem's requirements, the boxed answer is:

$\boxed{2}$

C.4. Overgeneralizing patterns

Here, FLASH-THINKING chooses an explicit polynomial to test out on problem 2, but then decides to generalize its conclusion to all polynomials.

FLASH-THINKING

Now, let us choose specific real roots for $P(x)$. Let $r_i = i$ for $i = 1, 2, \dots, n$. Let $P(x) = c(x-1)(x-2)\cdots(x-n)$ for some nonzero real number c. Then $P(x)$ is a polynomial of degree n with real coefficients, nonzero constant term, and no repeated roots.
Consider the divisor $Q(x) = (x-1)(x-2)\cdots(x-k) = \sum_{i=0}^{k} a_i x^i$.
Then $r_1 = 1, r_2 = 2, \dots, r_k = k$. For this choice, $a_i = (-1)^{k-i} e_{k-i}(1, 2, \dots, k)$.
Since $1, 2, \dots, k$ are all positive, $e_j(1, 2, \dots, k) > 0$ for $j = 1, 2, \dots, k$. Also $e_0(1, 2, \dots, k) = 1 > 0$.
Thus, for $i = 0, 1, \dots, k$, $a_i = (-1)^{k-i} e_{k-i}(1, 2, \dots, k) \neq 0$.
Therefore, for this divisor $Q(x) = a_k x^k + \cdots + a_0$, the product $a_0 a_1 \cdots a_k \neq 0$.
However, the given condition states that for any real numbers a_0, a_1, ..., a_k such that $a_k x^k + \cdots + a_1 x + a_0$ divides $P(x)$, the product $a_0 a_1 \cdots a_k$ is zero.
This is a contradiction.
Hence, our assumption that $P(x)$ has only real roots must be false.
Therefore, $P(x)$ has a nonreal root.

Final Answer: The final answer is $\boxed{P(x) \text{ has a nonreal root}}$

C.5. Hallucinating Citations

A common failure mode of GEMINI-2.5-PRO was its tendency to generate fake citations to support its points. One of the best examples is the case below, where the model hallucinates a paper with an emoji in the title.

GEMINI-2.5-PRO

We will use the following lemma, which is a known result in fair division theory. This result can be found, for example, as Lemma 7 in "Fair partitioning of 🍖" by Cardinal, Langerman, Palvolgyi (2021), or Theorem 2 in "Fair division of a discrete item" by Algaba et al. (2019). These papers attribute the proof technique to Gale (1993) and Scarf (1967).

Fair Partition Lemma: Let $v_1, \dots, v_m \in \mathbb{R}^n_{\geq 0}$. Suppose that for each $i \in \{1, \dots, n\}$, there exists a partition of $\{1, \dots, m\}$ into n sets $J_{i,1}, \dots, J_{i,n}$ such that $(\sum_{j \in J_{i,k}} v_j)_i \geq 1$ for all $k \in \{1, \dots, n\}$. Then there exists a partition of $\{1, \dots, m\}$ into n sets $K_1, \dots, K_n$ such that $(\sum_{j \in K_i} v_j)_i \geq 1$ for all $i \in \{1, \dots, n\}$.

C.6. Boxing solutions

The majority of models also struggled with presenting clear and structured solutions, making evaluation and understanding significantly more difficult. In the instance below, GEMINI-2.5-PRO boxed an entire solution at the end of its generation.

GEMINI-2.5-PRO

...
The representation is $(2, 16, 11)_{22}$. The digits are $a_0 = 11, a_1 = 16, a_2 = 2$. All are greater than $d = 1$. The proof holds.

Final Answer: The final answer is $\boxed{\text{Let B = } 2n\text{. We want to show that there exists a positive}}$
$\boxed{\text{integer } N \text{ such that for every odd ...}}$
$\boxed{\text{...As shown above, this implies } a_i > d \text{ for all digits } a_i \text{ in}}$
$\boxed{\text{the base-}2n \text{ representation of } n^k\text{. This completes the proof.}}$