
PROOF OR BLUFF? EVALUATING LLMs ON 2025 USA MATH OLYMPIAD

Ivo Petrov², Jasper Dekoninck¹, Lyuben Baltadzhiev², Maria Drencheva², Kristian Minchev²
Mislav Balunović^{1,2}, Nikola Jovanović¹, Martin Vechev^{1,2}

¹ETH Zurich

²INSAIT, Sofia University "St. Kliment Ohridski"

⊕ <https://matharena.ai/>

🔗 <https://github.com/eth-sri/matharena>

ABSTRACT

Recent math benchmarks for large language models (LLMs) such as MathArena indicate that state-of-the-art reasoning models achieve impressive performance on mathematical competitions like AIME, with the leading model, O3-MINI, achieving scores comparable to top human competitors. 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 the first 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 reveal that all tested models struggled significantly, achieving less than 5% on average. Through 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 suggest that current LLMs are inadequate for rigorous mathematical reasoning tasks, highlighting the need for substantial improvements 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 like MathArena (Balunović et al., 2025) demonstrate that these models achieve impressive performance on mathematical competitions such as AIME and HMMT. However, these competitions only evaluate final numerical answers and do not require rigorous proof-based reasoning essential for most mathematical tasks.

Current benchmarks that mitigate this issue either rely on formal verification tools like Lean (Zheng et al., 2022; Liu et al., 2023; Tsoukalas et al., 2024) or focus on the evaluation of constructive proofs (Balunović et al., 2025). While these approaches are useful, the former does not take advantage of LLMs' strong natural language generation capabilities, and the latter covers only a limited subset of proofs. Therefore, it remains uncertain whether LLMs can reliably address complex mathematical questions requiring rigorous reasoning, which are crucial in real-world mathematical contexts.

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 and well-explained solutions.

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

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 App. B

2 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.

2.1 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 App. 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 App. 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.

2.2 JUDGE SELECTION AND TRAINING

Our grading team consisted of four experts, each having substantial mathematical problem-solving experience as former national IMO team members or having participated in final-stage team selection processes for their countries. Prior to the grading, judges received instructions detailing evaluation goals and methodologies. These guidelines are accessible in our GitHub repository. 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.

2.3 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, modeled after the IMO’s evaluation process, ensures consistency in our grading and decreases personal 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 our GitHub repository and displayed on our website. An example can be found in App. 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 App. A.3.

Evaluators also documented prominent failure modes observed during grading. A "failure mode" was defined as the first instance of incorrect or inadequately explained reasoning, such as flawed logic, unjustified assumptions, mathematical inaccuracies, or computational mistakes. Specifically, mistakes were categorized into four classes:

Table 1: Main results of our evaluation. Problems are scored out of 7 points, with the total score being at most 42. Listed scores are averaged over all four runs. We measure cost in USD, and report the average score across all generations and graders for each problem.

Model	P1	P2	P3	P4	P5	P6	Total	Cost
R1	0.5	0.0	0.0	1.5	0.0	0.0	2.0	2.03
FLASH-THINKING	1.5	0.0	0.0	0.0	0.2	0.0	1.8	N/A
CLAUDE 3.7	0.5	0.5	0.0	0.0	0.0	0.6	1.5	9.03
QWQ	1.2	0.0	0.0	0.0	0.0	0.0	1.2	0.42
O1-PRO	0.5	0.0	0.0	0.0	0.2	0.4	1.2	203.44
O3-MINI	0.5	0.1	0.0	0.0	0.0	0.2	0.9	1.11

- **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 App. 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 §4.

3 RESULTS

We now present the results of our evaluation. In §3.1, we detail our primary findings, demonstrating that all models perform poorly on USAMO problems. In §3.2, we analyze common failure patterns in depth, identifying typical mistakes and trends in the models’ reasoning. Finally, in §3.3, we try and fail to automatically grade the models’ solutions by giving a judge model the grading scheme and a ground-truth solution.

3.1 MAIN RESULTS

We evaluate six state-of-the-art reasoning models on the 2025 USAMO problems. Specifically, we chose QWQ (Qwen, 2025), R1 (Guo et al., 2025), FLASH-THINKING (Reid et al., 2024), O1-PRO, O3-MINI (OpenAI, 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 App. A.1 for the full model names. The chosen hyperparameters and prompt can be found in App. 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.

While current state-of-the-art LLMs achieve performance comparable to top human competitors on numerical-answer-focused competitions such as AIME and HMMT, our evaluation uncovers a significant gap in their ability to generate rigorous proofs. The highest average score achieved by any evaluated model falls below 5%, indicating substantial limitations in handling the complexity and rigor of USAMO problems. Notably, among nearly 150 evaluated solutions from all models, none attained a perfect score. Although the USAMO presents more difficult problems compared to previously tested competitions, the complete failure of all models to successfully solve more than one problem underscores that current LLMs remain inadequate for rigorous olympiad-level mathematical reasoning tasks. Additionally, this limitation suggests that existing optimization methods like GRPO (Shao et al., 2024) may currently be insufficient for tasks requiring detailed logical precision.

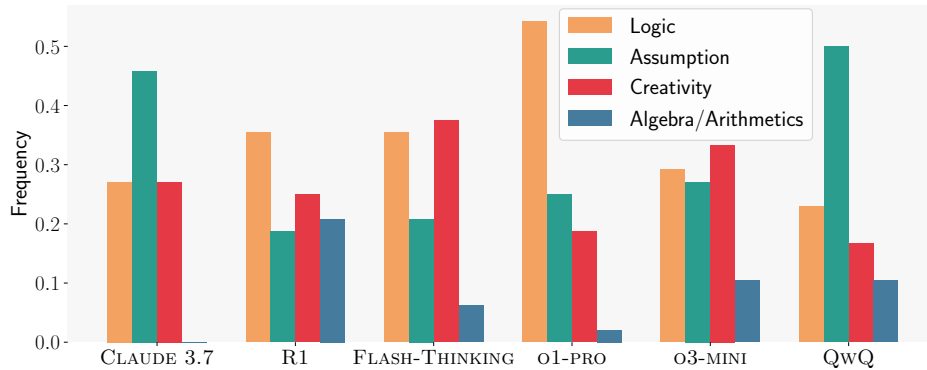


Figure 2: Distribution of first encountered failure mode.

3.2 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. This discrepancy poses a significant challenge for mathematical applications of LLMs as mathematical results derived using these models cannot be trusted without rigorous human validation. To further investigate this limitation, we conducted a thorough analysis of the errors identified during the grading process using the categories defined in §2.

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 or standard procedure without proper justification. Notably, O3-MINI, despite being one of the best reasoning models, frequently skipped essential proof steps by labeling 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 App. C.2.

However, the models generally demonstrated strong performance in algebraic and arithmetic computations, successfully performing symbolic manipulations without external computational support. Still, R1 showed a notably higher frequency of algebraic or arithmetic errors, indicating a clear area for targeted improvement in this model.

3.3 AUTOMATED GRADING

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

As detailed in Table 2, neither model 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 received significantly lower scores from the automated evaluation compared to other models. We hypothesize this discrepancy arises because FLASH-THINKING tends to generate multiple solutions per attempt, 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.

Table 2: Results of automated grading executed by O3-MINI and R1. The table shows the average total score which is at most 42.

Model	Ours	O3-MINI	CLAUDE 3.7
R1	2.0	19.3	14.9
FLASH-THINKING	1.8	10.5	14.1
CLAUDE 3.7	1.5	19.0	18.4
QWQ	1.2	23.8	18.8
O1-PRO	1.2	19.3	21.0
O3-MINI	0.9	19.5	17.1

4 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 problem statement. Consequently, it incorrectly insisted that the final answer was 2, even though it had otherwise correctly deduced that all even numbers satisfy the given conditions (see App. C.3). This behavior illustrates how alignment techniques like 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 a formal proof for such a claim (see App. C.4).

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. 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.

5 CONCLUSION

In this study, we comprehensively analyzed the performance of six state-of-the-art LLMs on problems from the USAMO 2025 competition. Using a rigorous human evaluation setup, we found that all evaluated models performed very poorly, with even the best-performing model achieving an average accuracy of less than 5%. Through detailed examination of the models’ reasoning traces, we identified several critical failure modes, including significant artifacts arising from the optimization strategies employed during model training. These findings underscore the substantial limitations of current LLMs in the rigorous mathematical reasoning required for high-level olympiad competitions, highlighting the need for substantial improvements in proof generation capabilities.

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. Math-construct: 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>.
- Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. Matharena: Evaluating llms on uncontaminated math competitions, February 2025. URL <https://matharena.ai/>.
- 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.
- 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.
- OpenAI. Openai o3-mini system card, January 2025. URL <https://cdn.openai.com/o3-mini-system-card-feb10.pdf>.
- 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, Zabeer 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, Amityay Thakur, and Swarat Chaudhuri. Putnambench: Evaluating neural theorem-provers on the putnam mathematical competition. *CoRR*, abs/2407.11214, 2024.
- 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:

- O3-MINI (HIGH) as O3-MINI
- O1-PRO (HIGH) as O1-PRO
- DEEPSEEK R1 as R1
- QWQ-32B as QWQ
- GEMINI-2.0-FLASH-THINKING-EXP as FLASH-THINKING
- CLAUDE-3.7-SONNET-THINKING as CLAUDE 3.7

A.2 HYPERPARAMETERS AND PROMPT

Model Hyperparameters For O3-MINI, O1-PRO, CLAUDE 3.7, and FLASH-THINKING, we used the default hyperparameters provided by their official API. For O3-MINI and O1-PRO, we used the high reasoning setting, which provides the highest available reasoning budget. For CLAUDE 3.7 and FLASH-THINKING, we set the maximum reasoning token limit to 32000.

As recommended by the authors of R1 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 models¹. The token limit was set to 32000. We use the TogetherAPI for both these 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}
```

```
## Grading scheme
```

```
{grading_scheme}
```

```
# Example
```

¹See <https://huggingface.co/deepseek-ai/DeepSeek-R1> and <https://huggingface.co/Qwen/QwQ-32B>

```

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 App. 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 = floor(n^k / (2n)^i) 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 >= floor(n / 2^i) ||."
    },
    {
      "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  $a_i = \left\lfloor \frac{n^k}{(2n)^i} \right\rfloor \bmod(2n)$ "
    },
    {
      "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  $a_i \geq \left\lfloor \frac{n}{2^i} \right\rfloor$ ."
    },
    {
      "title": "Finishing",
      "points": 1,
      "desc": "Chooses a provably sufficiently large N and has proven that  $n^k$  has more than k digits."
    }
  ]
}
```

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, \dots, a_k such that the polynomial $a_k x^k + \dots + a_1 x + a_0$ divides $P(x)$, the product $a_0 a_1 \dots a_k$ 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 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 $R \in S$ 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 $\frac{1}{n+1} \sum_{i=0}^n \binom{n}{i}^k$ is 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 $r_m, r_l \notin \{s_1, s_2, \dots, s_{k-1}\}$), 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 $j = 1, 2, \dots, k$, 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" $S \subset \mathbb{R}^2$, 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 ω) lies on the perpendicular bisector of any chord, we will see that the perpendicular from the center of ω to BC must pass through C .

...

C.1.4 EXAMPLE OF ALGEBRAIC/ARITHMETIC ERROR

Here, 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.

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 $S \rightarrow P_i$ capacity 1. Edge $P_i \rightarrow G_{i,k}$ capacity 1. Edge $G_{i,k} \rightarrow C_j$ if $C_j \in G_{i,k}$ capacity 1. Edge $C_j \rightarrow T$ 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 $S \rightarrow P_i$ capacity 1 for $i = 1, \dots, n$. Edge $P_i \rightarrow C_j$ capacity 1 for $i = 1, \dots, n, j = 1, \dots, m$. Edge $C_j \rightarrow T$ 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 $S \rightarrow P_i \rightarrow C_j \rightarrow T$. Let x_{ij} be the flow on edge (P_i, C_j) . Then $\sum_{j=1}^m x_{ij} \leq 1, \sum_{i=1}^n x_{ij} \leq 1, \sum_{i=1}^n \sum_{j=1}^m x_{ij} = n$. 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 , $\sum_{C_j \in M \text{ and } (P_i, C_j) \in M} s_{ij} \geq 1$. 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 $\sum_{C_j \in F} s_{ij} \geq 1$. 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 $i \neq j$. 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 $i \neq j$.

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 $G_{i,j}$. 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 $k = 1, \dots, n$, $\sum_{C_j \in G_{i,k}} s_{ij} \geq 1$.

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 $X \subseteq U$, $|N(X)| \geq |X|$, where $N(X) = \bigcup_{P_i \in X} \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}$.

For any $X \subseteq U$,

$|N(X)| = |\bigcup_{P_i \in X} \{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}| \leq \sum_{P_i \in X} |\{G_{i,1}, G_{i,2}, \dots, G_{i,n}\}| = \sum_{P_i \in X} n = n|X|$. 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

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 for any positive integer m .

However, since the problem requires a boxed answer and the minimal solution is typically expected, the answer is . But strictly speaking, the correct answer is all even positive integers. Given the problem's requirements, the boxed answer is:

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, \dots, 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 $P(x)$ has a nonreal root