

# MuMath: Multi-perspective Data Augmentation for Mathematical Reasoning in Large Language Models

Anonymous ACL submission

## Abstract

Recently, the tool-use Large Language Models (LLMs) that integrate with external Python interpreters have significantly enhanced mathematical reasoning capabilities for open-source LLMs. However, these models fall short in demonstrating the calculation process, which compromises user-friendliness and understanding of problem-solving steps. Conversely, while tool-free methods offer a clear display of the problem-solving process, their accuracy leaves room for improvement. These tool-free methods typically employ a somewhat narrow range of augmentation techniques such as rephrasing and complexity enhancement to boost performance. In response to this issue, we have amalgamated and further refined these strengths while broadening the scope of augmentation methods to construct a multi-perspective augmentation dataset for mathematics—termed **MuMath** ( $\mu$ -Math) Dataset. Subsequently, we finetune LLaMA-2 on the MuMath dataset to derive the MuMath model. Our experiments indicate that our MuMath-70B model achieves new state-of-the-art performance among tool-free methods—achieving 84.5% on GSM8K (an increase of 2.2% compared to the previous best open-source LLM) and 32.2% on MATH (a rise by 5.6% compared to the prior best open-source LLM). We release the MuMath dataset along with its corresponding models and code for public use.

## 1 Introduction

Large Language Models (LLMs) (Devlin et al., 2019; Radford et al., 2019; Liu et al., 2019; Brown et al., 2020; Raffel et al., 2023), especially proprietary LLMs like GPT-4 (OpenAI, 2023b), have been proven to be predominant across almost all the tasks in Natural Language Processing (NLP), including text classification (Jiang et al., 2023b; Min et al., 2022), code generation (Chen et al., 2021;

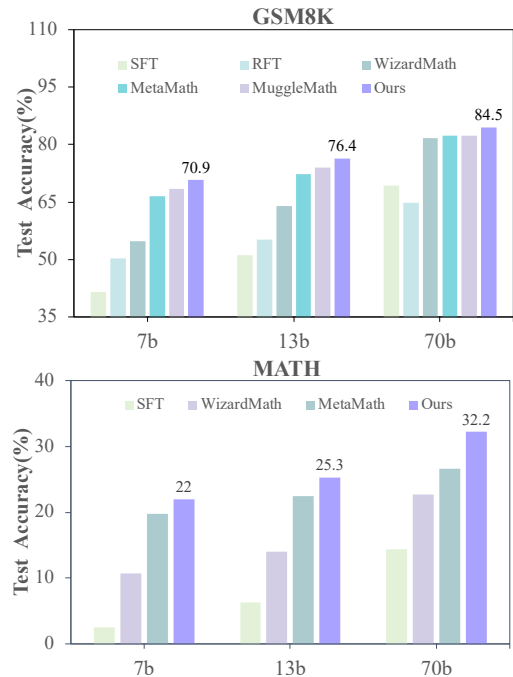


Figure 1: Comparing MuMath with baselines on LLaMA-2 base models from 7B to 70B, it’s observed that MuMath demonstrate significant enhancement over previous state-of-the-art mathematical reasoning LLMs. Remarkably, against MetaMath on the MATH dataset, MuMath improves by a margin of 5.6%.

Luo et al., 2023b), instruction following (Longpre et al., 2023), and mathematical reasoning (Li et al., 2023; Yu et al., 2023; Gou et al., 2023). Among these, mathematical ability is an important and typical aspect for evaluating different LLMs, and there still remains a considerable gap between open-source LLMs, e.g., LLaMA (Touvron et al., 2023), and the proprietary LLMs in the realm of mathematical problem solving (Yue et al., 2023).

Recently, a multitude of studies dedicated to enhancing the mathematical capabilities of open-source LLMs, which can be generally divided into two different research trajectories: tool-use and tool-free. As for the tool-use LLMs, they are typically integrated with external Python interpreters, making full use of the latter’s impeccable abili-

ties in numerical calculation and logical inference which can substantially assist LLMs in solving complex mathematical problems, e.g., PAL (Gao et al., 2023), PoT (Chen et al., 2023), MAMmoTH (Yue et al., 2023), TORA (Gou et al., 2023) and MathCoder (Wang et al., 2023). Although the tool-use method can solve computational errors through code, it lacks a demonstration of the calculation process, making it less user-friendly in terms of understanding the problem-solving steps. On the other hand, while the tool-free method provides a good display of the problem-solving process, its accuracy still needs to be improved. Therefore, our work follows along the tool-free trajectory, focusing on improving the math reasoning ability of LLMs.

Representative tool-free methods adopt supervised finetuning (SFT) on the augmented datasets to enhance the LLMs’ mathematical reasoning capability, including RFT (Yuan et al., 2023), MetaMath (Yu et al., 2023), WizardMath (Luo et al., 2023a), and MuggleMath (Li et al., 2023), etc. RFT only augments the answer via rejection sampling to produce diverse reasoning paths with correct answers, but the generated data is similar to training dataset. MetaMath utilizes two simple augmentation methods, that one uses rephrasing to enhance the narrative diversity of the questions and answers, and the other adopts the SV (Weng et al., 2023) and FOBAR (Jiang et al., 2023a) to generate new mathematical problems and problem-solving strategies for equations. Instead of rephrasing, WizardMath and MuggleMath create new questions via rephrasing and complexity enhancement, thus apparently improving the diversity of the dataset. However, the augmenting perspectives of these two methods are not sufficiently comprehensive, and the accuracy rate of the answers to new questions is suboptimal.

While their constructed augmented dataset enhances the capability of the model, different works adopt different methods and employ a rather limited variety of augmentation methods. So we integrate and further enhance their strengths and expand the perspective of augmentation methods to construct a **multi-perspective** augmentation dataset for **math**, called **MuMath** ( $\mu$ -Math) Dataset, including four categories. (1) In Data Reformulation, besides the question rephrasing, we propose the solution reorganization to provide a comprehensive roadmap for the process and detailed answers. (2) In Backward Creation, We have retained the FOBAR method and introduced the

Backward-Forward Transformation (BF-Trans) approach, which transforms equation-solving into arithmetic problem-solving, generating new problems and solution methods that are distinctly different from the FOBAR style. (3) We’ve further refined the existing question alteration from a fresh perspective: expression replacement. It offers a controllable and innovative way, compared to simply changing numbers or arbitrarily increasing complexity. Also, we utilize majority sampling finetuning to boost answer accuracy and data quality. (4) Additionally, beyond data augmentation for mathematical problem solving, we propose a Nested Multi-task Construction Augmentation, where we nest plan programming or question summarizing texts into the solution, combining data of auxiliary tasks into the main task as solving the math problem. Through the process of supervised fine-tuning on open-source language models, such as LLaMA-2, and applying it to the MuMath dataset, we have successfully developed MuMath models in a variety of sizes. This demonstrates that the dataset has the potential to significantly enhance the mathematical capabilities of open-source models.

Our contributions are as follows:

- We construct a multi-perspective augmentation dataset for math, called MuMath Dataset, including data reformulation, backward creation, question alteration and nested multi-task.
- We conducted extensive experiments to demonstrate the effectiveness of different augmentations, as well as give some insights on mathematical reasoning for LLMs.
- By supervised fine-tuning on the open-source LLMs on the MuMath dataset, we obtain the MuMath model, which achieves new state-of-the-art performances among tool-free methods. MuMath-70B has achieved 84.5% on GSM8K (Cobbe et al., 2021) (+2.2% compared to the previous best open-source LLM) and 32.2% on MATH (Hendrycks et al., 2021a) (+5.6% compared to the previous best open-source LLM).

## 2 Related Work

**Mathematical Reasoning** Currently, there are two main research trajectories to enhance the mathematical ability of open-source models. (1) The first trajectory focuses on LLMs purely, without tool use. Yuan et al. (2023) propose a representative

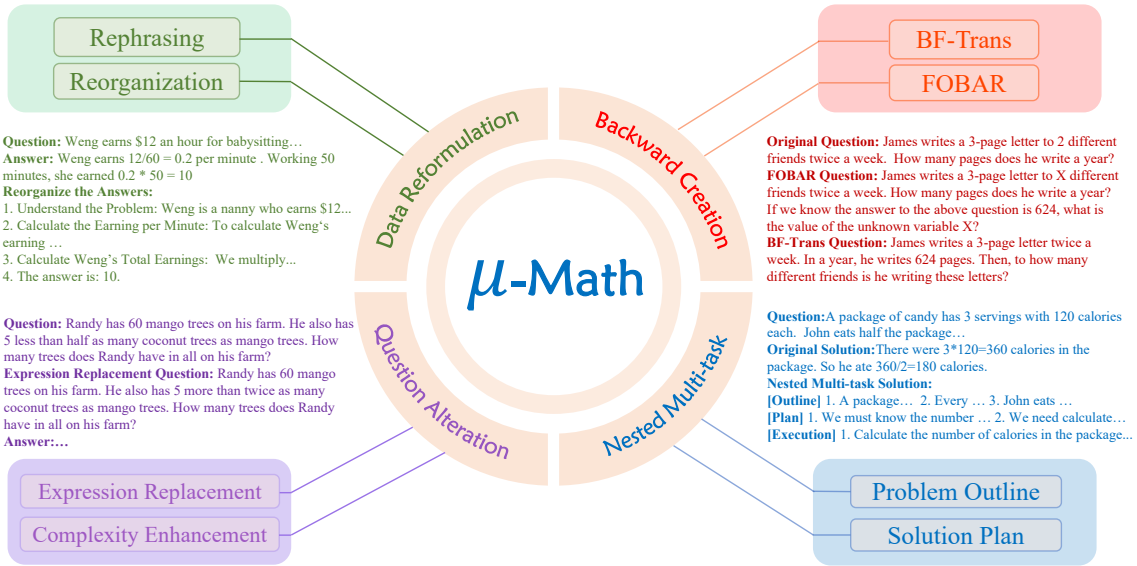


Figure 2: Overview of the augmentation methods our MuMath employs, which can be divided into four categories: (1) Data Reformulation includes solution reorganization and question rephrasing; (2) Backward Creation includes Backward-Forward Transformation (BF-Trans) and FOBAR; (3) Question Alteration includes expression replacement and complexity enhancement; (4) Nested Multi-task construction includes data of the auxiliary tasks, i.e., Problem Outline and Solution Plan. Please zoom in the image for a better view.

160 tool-free methods, leveraging rejection sampling  
161 finetuning (RFT) to enhance Llama’s mathemat-  
162 ical ability, while WizardMath (Luo et al., 2023a)  
163 chooses a reinforcement learning (RL) framework  
164 and evolves its math capability through proximal  
165 policy optimization (PPO, Schulman et al.,  
166 2017). The most recent tool-free methods are Mug-  
167 gleMath (Li et al., 2023) and MetaMath (Yu et al.,  
168 2023), both of which manage to augment math  
169 problem-solution data followed by finetuning the  
170 open LLMs on these newly acquired data. (2) The  
171 second trajectory underscores the integration of  
172 LLMs with tool use, with Program-aided Language  
173 model (PAL, Gao et al., 2023) and Program of  
174 Thought (PoT, Chen et al., 2023) being two pion-  
175 ering works. Besides, MAMMO TH (Yue et al., 2023)  
176 employs both CoT and PoT that are combined in  
177 a coarse-grained manner, with different samples  
178 utilizing different approaches. TORA (Gou et al.,  
179 2023) performs imitation learning with problem so-  
180 lutions organized by interleaving multiple steps of  
181 program-based tool use and natural language rea-  
182 soning. Similarly, MathCoder (Wang et al., 2023),  
183 contemporaneous with TORA, also curates the rea-  
184 soning text in an interleaving manner, while it in-  
185 troduces interpolation problems to bridge the gap  
186 in difficulty between GSM8K (Cobbe et al., 2021)  
187 problems and MATH (Hendrycks et al., 2021b)

188 problems. Our work concentrates on fully exploit-  
189 ing LLM’s pretrained capability to conduct mathe-  
190 matical reasoning, thus progressing along the tool-  
191 free research trajectory.

192 **Data Augmentation** The process of data aug-  
193 mentation for mathematical reasoning can be di-  
194 vided into two categories. The first one involves  
195 enhancing the content of both the problem and  
196 its solution without altering their original mean-  
197 ing. CoT approaches (Wei et al., 2022; Fu et al.,  
198 2022) present the solution in a step-by-step format,  
199 making it easier for the model to learn. Rephras-  
200 ing (Yu et al., 2023; Li et al., 2023) refers to re-  
201 stating the problem and its solution in a different  
202 manner, thereby generating a new sequence of to-  
203 kens. Rejection sampling (Yuan et al., 2023) uti-  
204 lizes a fully trained model to yield new reasoning  
205 paths, and the quality of these paths is intrinsically  
206 linked to the performance of the trained model.  
207 The other one modifies the computational values  
208 or logic of a problem, thus generating new prob-  
209 lems. Then, through the method of knowledge  
210 distillation (Huang et al., 2022; Li et al., 2022;  
211 Magister et al., 2023; Ho et al., 2023; Fu et al.,  
212 2023b; Shridhar et al., 2023), it generates new so-  
213 lutions and transfers reasoning abilities from the  
214 teacher model (for instance, GPT4). The Evol-  
215 instruct method (Xu et al., 2023; Luo et al., 2023a)

### Example 3.1: Reorganization

**Question:** Weng earns \$12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn?

**Answer:** Weng earns  $12 / 60 = 0.2$  per minute . Working 50 minutes, she earned  $0.2 * 50 = 10$ .

#### Reorganization Answers:

1. **Understand the Problem:** - Weng is a babysitter who earns \$12 per hour. Yesterday, she worked for 50 minutes. Our task is to calculate how much she earned.

2. **Calculate the Earning per Minute:** - To calculate Weng’s earning per minute, we divide her hourly earning by 60 (the total number of minutes in an hour):  $12 / 60 = 0.2$  per minute.

3. **Calculate Weng’s Total Earnings:** - We multiply Weng’s earning per minute by the number of minutes she worked:  $0.2 * 50 = 10$ .

4. **Answer:** - Weng earned 10 for her 50 minutes of babysitting. **The answer is: 10.**

and complexity enhancement (Li et al., 2023) incorporate modifications such as adding constraints, adjusting the context, and more to the original data. Fobar (Jiang et al., 2023a) generates a series of questions for backward reasoning by masking numbers. It then samples a set of backward reasoning chains to predict the masked number. Our proposed method not only enriches these two types of augmentation, but also adds a multi-task augmentation category. This can be nested into the existing data to bolster the mathematical reasoning capabilities of the model.

## 3 Methods

The overview of our method is illustrated in Figure 2. The prompts are in Appendix B.

### 3.1 Data reformulation

Our data reformulation can be divided into two primary categories: reorganization and rephrasing.

**Reorganization** Inspired by the learning of human cognitive process, it becomes evident that detailed and standardized solutions are easier to comprehend. We have made solving steps significantly more understandable for learning by incorporating reorganization. Phrases such as "understand the problem", "define variables", and "calculate the number" act as explicit instructions, leading us toward the final result by "The answer is". See Example 3.1 for details. Delving into this process, reorganization further divides each step into more specific components, effectively transforming an intricate question into a series of simpler ones. Each mini-question formed from this segmentation is then addressed thoroughly. This approach not only reduces the complexity of each solving step but also significantly mitigates learning difficulties. We use  $S_{reorg}$  to denote the reconstructed solution,

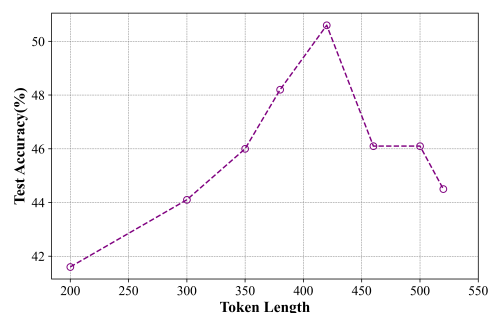


Figure 3: The relationship between token length and accuracy on GSM8K test set.

and thus the new dataset we get can be formalized as  $\mathcal{D}_{reorg} = \{(Q, S_{reorg})\}$ .

For the reorganization solutions, we manipulated response length by adding a minimum word count restriction in the prompt. Upon examining the generated response, it was discerned that longer token lengths corresponded to lower complexity in overall responses. However, the parsing steps become redundant when the token length becomes excessively long. The result could potentially lead to models assimilating irrelevant information while overlooking correct answers. See the example in Appendix A.1. Consequently, this underscores the importance of optimal response length for ensuring model efficacy during reorganization augmentation. So we fine-tune LLaMA-2 7B utilizing data of varying token lengths and subsequently depict the correlation between token length and accuracy. Figure 3 shows a linear accuracy increase for token lengths between 200 and 420, but the accuracy begins to decline when the token length exceeds 420. So we have chosen to utilize a token length of approximately 420 for the reorganization data.

**Rephrasing** While reorganization merely amplifies solutions, rephrasing augments questions without altering the original intent. After requesting

278 answers to the rephrasing questions, we can get  
279  $\mathcal{D}_{reph} = \{(Q_{reph}, S_{reph})\}$  by filtering out ques-  
280 tions with incorrect answers. Combining these 2  
281 datasets, we have the reformulation dataset  $\mathcal{D}_1 =$   
282  $\mathcal{D}_{reorg} \cup \mathcal{D}_{reph}$ .

### 283 3.2 Backward-Forward Transformation

284 FOBAR masks some specific value in the origi-  
285 nal forward question using “X”, convert the final  
286 answer to a new condition, and thus construct a  
287 backward question by asking to find the unknown  
288 variable X. However, this method tends to list equa-  
289 tions concerning X and then solve them, as still a  
290 forward reasoning process. Here our purpose is to  
291 introduce backward questions with directly arith-  
292 metic solutions instead of equation solving, i.e.,  
293 engage in as much reverse reasoning as possible.

294 To this end, we propose a new method called  
295 **Backward-Forward Transformation (BF-Trans)**.  
296 For a certain question-answer pair, we firstly utilize  
297 FOBAR to transform the original question  $Q$  into  
298 a backward one  $Q_b$ ; secondly, we rephrase the FO-  
299 BAR question into a new form where the masked  
300 value is requested directly instead of employing  
301 an unknown variable X, resulting in a “secondary  
302 forward” question which we called BF-Trans ques-  
303 tion, marked as  $Q_{bf}$ . Example 3.2 shows the  
304 differences among the original question, FORAR  
305 and BF-Trans. Finally, we generate the solution  
306  $S_{bf}$  for this BF-Trans question. Collecting all  
307 these BF-Trans augmented samples, we can have  
308  $\mathcal{D}_{bf} = \{(Q_{bf}, S_{bf})\}$ . Note that the final answer of  
309 the BF-Trans solution is correct after the filtering  
310 procedure, corresponding to a certain masked num-  
311 ber of the FOBAR question is corresponding to a  
312 certain number.

313 Combined with the FOBAR dataset  $\mathcal{D}_{fobar}$ ,  
314 hence the backward reasoning part of our final train-  
315 ing set is  $\mathcal{D}_2 = \mathcal{D}_{bf} \cup \mathcal{D}_{fobar}$ .

### 316 3.3 Question Alteration

317 Our observations have highlighted that diversity  
318 and complexity inherent within training data play  
319 an instrumental role in enhancing mathematical  
320 reasoning capabilities. So we also strive to en-  
321 hance our model’s ability to generalize by gener-  
322 ating brand new problems. We have employed a  
323 more diversified perspective in generation and sig-  
324 nificantly enhanced the quality of our data.

325 **Complexity Enhancement** Drawing inspiration  
326 from (Luo et al., 2023b) and (Li et al., 2023), we in-

crease problem complexity to create new questions  
 $Q_{complex}$ . Our methods include but are not limited  
to adding constraints and modifying context.

**Expression Replacement** We assert that chang-  
ing numerals doesn’t alter the logic of the calcu-  
lation, representing a singular enhancement. Con-  
versely, arbitrarily increasing complexity is exces-  
sively unrestricted. Thus, to broaden our perspec-  
tive on question alteration, we introduce expression  
replacement as a novel and controlled alteration  
method that has a different calculation logic intrin-  
sically. This method offers an interpolated perspec-  
tive between changing numerical and increasing  
complexity arbitrarily. The comparison of these  
three methods can be found in the Appendix A.2.  
Upon careful examination of the problem-solving  
process, it becomes evident that the mathematical  
expression plays an integral role. If we change  
multiplication to division in an equation, it signif-  
icantly shifts the problem’s intent, requires a dif-  
ferent computational logic, and generates a new  
question. Our approach operates in this manner:  
we first extract all mathematical expressions from  
the solution. Subsequently, an arithmetic expres-  
sion is altered to form a novel equation. With the  
original problem statement and new equations as  
guides, a new question can be generated denoted  
as  $Q_{replace}$ . Example 3.2 compares the original  
questions and the expression replaced one.

**Majority Sampling Finetuning** After generat-  
ing new questions, we utilized GPT4 for solutions.  
A challenge emerges as these new questions lack  
standard reference answers, possibly introducing  
errors into the training data. Despite this, our exper-  
iments showed satisfactory performance from mod-  
els trained with this data. We hypothesize that cor-  
rect steps within incorrect final answers might as-  
sist LLMs in understanding math problems, align-  
ing with theories proposed in (Fu et al., 2023a) and  
(Yu et al., 2023). To maximize answer accuracy  
for new questions, we implemented Majority Solu-  
tion Sampling to achieve a higher-accuracy dataset  
for these queries. We utilize majority voting with  
 $k = 30$  to request solutions and only select one re-  
sponse with the majority answer for finetuning. We  
name the above procedure as Majority Sampling  
Finetuning (MSF).

We use  $S_{replace}$  and  $S_{complex}$  to stand for  
the generated solutions to the newly intro-  
duced questions  $Q_{replace}$  and  $Q_{complex}$  respec-  
tively, resulting in our recreation dataset  $\mathcal{D}_3 =$

### Example 3.2: FOBAR vs BF-Trans

**FOBAR Question:** Joe’s mother gave him \$56 to go to the store. Joe bought  $X$  notebooks and 2 books. Each notebook costs \$4 and each book costs \$7. How much money does Joe have left? If we know the answer to the above question is 14, what is the value of the unknown variable  $X$ ?

**FOBAR Response:** We know Joe have \$14 left after he bought  $X$  notebooks and 2 books. Each notebook costs \$4 and he bought  $X$  of them, so he spent  $4X$  dollars on notebooks. Each book costs \$7 and he bought 2 of them, so he spent  $2 \times 7 = 14$  dollars on books. Hence the total money he spent is  $4X + 14$ . We know he originally had \$56, so we have  $56 - (4X + 14) = 42 - 4X$  left. Since we know the answer is 14, we have  $42 - 4X = 14$ . **Solving for  $X$ ,** we get  $4X = 28$  and  $X = 7$ .

**BF-Trans Question:** Joe’s mother gave him \$56 to go to the store. Joe bought 2 books and some notebooks. Each notebook costs \$4 and each book costs \$7. If Joe has \$14 left after shopping, how many notebooks did he buy?

**BF-Trans Response :** Joe initially had \$56, and now has \$14 left after shopping, so he spent  $56 - 14 = 42$  dollars. Each book costs \$7 and he bought 2 books, so he spent  $7 \times 2 = 14$  dollars on books. He spent 42 dollars in total and 14 dollars on books, so he spent  $42 - 14 = 28$  dollars on notebooks. Each notebook costs \$4, so he bought  $28 / 4 = 7$  notebooks.

### Example 3.3: Expression Replacement

**Question:** Randy has 60 mango trees on his farm. He also has 5 less than half as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

**Response:** Half of the number of Randy’s mango trees is  $60 / 2 = 30$  trees. So Randy has  $30 - 5 = 25$  coconut trees. Therefore, Randy has  $60 + 25 = 85$  trees on his farm. The answer is: 85

**New Question:** Randy has 60 mango trees on his farm. He also has 5 more than twice as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

**New Response:** Twice the number of mango trees on Randy’s farm is  $60 \times 2 = 120$  trees. The total number of coconut trees on Randy’s farm is 5 more than twice the number of mango trees, a total of  $120 + 5 = 125$  trees. Altogether, Randy has  $125 + 60 = 185$  trees on his farm. The answer is: 185

$$\{(Q_{replace}, S_{replace})\} \cup \{(Q_{complex}, S_{complex})\}.$$

## 3.4 Nested Multi-task Learning

Multitask learning (Raffel et al., 2023; Sun et al., 2019) equips a single model with the capability to handle diverse tasks, and it can also enhance the main task processing ability of the model, by introducing strongly correlated auxiliary tasks. Different from continual learning (Parisi et al., 2019) where different tasks are separated in stage level (thus coarse-grained), multitask learning is a fine-grained procedure, and it integrates the data from different tasks into a single training batch for simultaneously learning (different tasks are distinguished in batch level). We propose a more fine-grained multi-task learning strategy called Nested Multi-Task learning (NestedMT), where we nest the data of auxiliary tasks into the data of the main task in a sample level.

Specifically, for the main task of solving mathematical problems  $Q$ , we select two auxiliary tasks: summarizing the question and listing the solving plan. Different from the stage-level and batch-level counterparts, we prepend the text of question outline  $O$ , solving plan  $P$ , or both to the solution text  $S$ , assembling into an individual final solution  $S_{mt} = O \oplus P \oplus S$ , where  $\oplus$  represents concatenation, for each original question. See the example in

Appendix A.3. Then we have  $\mathcal{D}_4 = \{(Q, S_{mt})\}$  as the nested multi-task dataset. In nested multi-task learning, our model can learn to solve the math problems and meanwhile learn to manage various auxiliary tasks strongly related to the math problem solving task itself. All these tasks are concentrated into one single sample and thus the auxiliary tasks can contribute in a more detailed and precise manner to improve the model’s performance on its principal task as math problem solving.

## 4 Experiments

### 4.1 Experimental Setup

**Datasets** We employ two widely recognized mathematical reasoning benchmarks. The first one, GSM8K (Cobbe et al., 2021), is a collection of high-quality elementary school math problems, comprising 7,473 training instances and 1,319 test instances. The second benchmark is the MATH dataset (Hendrycks et al., 2021a), which encompasses seven subjects, i.e., Prealgebra, Algebra, Number Theory, Counting and Probability, Geometry, Intermediate Algebra and Precalculus. This dataset includes math competition problems from high school level with a total of 7,500 training samples and 5,000 testing samples.

We employ a series of augmentation methods mentioned in Section 3, to create different subsets

based on the original GSM8K and MATH training data. Note that there are significant differences in difficulty levels and numbers of conditions between questions of these two datasets. Therefore, after requesting new solutions and the subsequent filtering, the amounts of data we obtained from GSM8K and MATH are slightly different.

**Implementation Details** Our study utilizes the state-of-the-art open-source LLMs for fine-tuning, comprising LLaMA-2 7B, LLaMA-2 13B, and LLaMA-2 70B (Touvron et al., 2023). All these models undergo full fine-tuning. We incorporate system prompts from (Taori et al., 2023) during the fine-tuning, and employ AdamW for optimization. We set the global batch size to 128 and used a cosine learning rate scheduler with a 0.03 warm-up period for 3 epochs. The computational hardware are NVIDIA A800 GPUs.

## 4.2 Results

### 4.2.1 Main results

In Table 1, we contrast the performance of current colsed-source LLMs, tool-use LLMs, and tool-free LLMs on GSM8K and MATH. It’s evident that MuMath set a new standard in the 7B LLMs. Compared to the baseline LLaMA-2 SFT, MuMath shows significant accuracy increases on GSM8K and MATH by 29.3% and 17.6%, respectively. In contrast to MetaMath, MuMath improves by 4.4% and 2.2% on GSM8K and MATH respectively. In LLMs with 13B parameters, MuMath surpasses MetaMath by 4.1% and 2.9% on GSM8K and MATH datasets respectively. For LLMs with 70B parameters, MuMath surpasses MetaMath by 2.2% on the GSM8K dataset. Significantly, against MetaMath on the MATH dataset, MuMath improves impressively by a margin of 5.6%. Note that our MuMath dataset contains approximately **274K** samples, apparently less than that of MetaMathQA (**390K**). This highlights our proposed data augmentation methods’ effectiveness in enhancing mathematical reasoning capabilities.

### 4.2.2 Ablation of Different Augmentation

In this section, we conduct experiments to study the effect of augmentations in MuMath. Table 2 shows the fine-tuning results of each sub-component within our proposed augmentation methods, tested on both GSM8K and Math datasets. The data size of each subset is consistent with the original data (7K). Each dataset shows substantial improvement

Model	GSM8K	MATH
<i>colsed-source LLMs</i>		
GPT-4 (OpenAI, 2023b)	92.0	42.5
GPT-3.5-Turbo (OpenAI, 2023a)	80.8	34.1
PaLM (540B)(Chowdhery et al., 2022)	56.5	8.8
PaLM-2 (540B) (Anil et al., 2023)	80.7	34.3
Minerva (540B) (Lewkowycz et al., 2022)	58.8	33.6
<i>tool-use LLMs</i>		
<i>7B</i>		
CodeLLaMa(PAL) (Rozière et al., 2023)	34.0	16.6
MAmmoTH (Yue et al., 2023)	53.6	31.5
MathCoder-L (Wang et al., 2023)	64.2	23.3
TORA (Gou et al., 2023)	68.8	40.1
<i>13B</i>		
CodeLLaMa(PAL) (Rozière et al., 2023)	39.9	19.9
MAmmoTH (Yue et al., 2023)	62.0	34.2
MathCoder-L (Wang et al., 2023)	72.6	29.9
TORA (Gou et al., 2023)	72.7	43.0
<i>70B</i>		
MAmmoTH (Yue et al., 2023)	76.9	41.8
MathCoder-L (Wang et al., 2023)	83.9	45.1
TORA (Gou et al., 2023)	84.3	49.7
<i>tool-free LLMs</i>		
<i>7B</i>		
LLaMA-2 (Touvron et al., 2023)	14.6	2.5
LLaMA-2 SFT (Touvron et al., 2023)	41.6	-
LLaMA-2 RFT (Yuan et al., 2023)	50.3	-
WizardMath (Luo et al., 2023a)	54.9	10.7
MetaMath (Yu et al., 2023)	66.5	19.8
MuggleMath (Li et al., 2023)	68.4	-
<b><math>\mu</math>-Math</b>	<b>70.9</b>	<b>22.0</b>
<i>13B</i>		
LLaMA-2 (Touvron et al., 2023)	24.3	6.3
LLaMA-2 SFT (Touvron et al., 2023)	51.1	9.2
LLaMA-2 RFT (Yuan et al., 2023)	55.3	-
WizardMath (Luo et al., 2023a)	63.9	14
MetaMath (Yu et al., 2023)	72.3	22.4
MuggleMath (Li et al., 2023)	74	-
<b><math>\mu</math>-Math</b>	<b>76.4</b>	<b>25.3</b>
<i>70B</i>		
LLaMA-2 (Touvron et al., 2023)	57.8	14.4
LLaMA-2 SFT (Touvron et al., 2023)	69.3	14.9
LLaMA-2 RFT (Yuan et al., 2023)	64.8	-
WizardMath (Luo et al., 2023a)	81.6	22.7
MetaMath(Yu et al., 2023)	82.3	26.6
MuggleMath (Li et al., 2023)	82.3	-
<b><math>\mu</math>-Math</b>	<b>84.5</b>	<b>32.2</b>

Table 1: Comparison of testing accuracy to existing LLMs on GSM8K and MATH. The fine-tuned data was merged with the augmented GSM8K and MATH, with respective data sizes of 127k and 147k.

Method	GSM8K		MATH	
	Datasize	Acc	Datasize	Acc
SFT	7K	41.6	7K	4.4
Reorganization	7K	50.6	7K	6.0
Rephrasing	7K	46.2	7K	5.9
Reorganization + Rephrasing	7K+7K	52.1	7K+7K	7.3
FOBAR	7K	40.6	7K	4.9
BF-trans	7K	42.8	7K	5.8
FOBAR + BF-Trans	7K+7K	46.2	7K+7K	7.4
Expression Replacement (ER)	7K	47.7	7K	6.4
Complexity Enhancement (CE)	7K	45.1	7K	4.6
ER + CE	7K+7K	48.5	7K+7K	7.0
Nested Multi-task	7K	51.0	7K	6.8
Separate Multi-task	7K+7K	42.5	7K+7K	6.6

Table 2: Different data augmentation strategies on GSM8K and MATH performances.

GSM8K					MATH				
$\mathcal{D}_1$	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$	Acc.	$\mathcal{D}_1$	$\mathcal{D}_2$	$\mathcal{D}_3$	$\mathcal{D}_4$	Acc.
50K	30K	40K	7K		50K	40K	50K	7K	
✓	✗	✗	✗	59.6	✓	✗	✗	✗	10.5
✗	✓	✗	✗	53.3	✗	✓	✗	✗	10.7
✗	✗	✓	✗	57.7	✗	✗	✓	✗	17.9
✗	✗	✗	✓	51.0	✗	✗	✗	✓	6.8
✓	✓	✗	✗	64.0	✓	✓	✗	✗	14.5
✓	✗	✓	✗	64.5	✓	✗	✓	✗	19.1
✓	✗	✗	✓	60.8	✓	✗	✗	✓	10.8
✗	✓	✓	✗	62.2	✗	✓	✓	✗	20.2
✗	✓	✗	✓	55.6	✗	✓	✗	✓	12.6
✗	✗	✓	✓	60.1	✗	✗	✓	✓	18.6
✓	✓	✓	✗	67.9	✓	✓	✓	✗	21.1
✓	✓	✗	✓	65.1	✓	✓	✗	✓	14.8
✓	✗	✓	✓	64.0	✓	✗	✓	✓	20.1
✗	✓	✓	✓	63.2	✗	✓	✓	✓	20.6
✓	✓	✓	✓	<b>69.2</b>	✓	✓	✓	✓	<b>21.6</b>
MetaMath				64.4	-				17.7
MuggleMath				68.4	-				-

Table 3: Effect of different data subsets on the accuracy of GSM8K and MATH.  $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$  and  $\mathcal{D}_4$  are data reformulation, backward creation, question alteration, and nested multi-task learning. We also compare our MuMath model with two baselines, all of which are trained on datasets augmented from only one source, i.e., only GSM8K or only MATH.

compared to the original data. Remarkably, the nested multi-task augmentation records a 9.4% increase under equal quantities on GSM8K. To sum up, all of our augmentation methods effectively boost the mathematical reasoning abilities of open-source LLMs. Moreover, from the results obtained by the stacked data, we discovered that the sub-methods within each of the four data augmentation methods are complementary to each other.

Table 3 enumerates the data volumes of four augmentation datasets, and it mainly presents the test accuracy of various augmentation combinations. As observed, the models trained on any kind of augmentations outperform the SFT method significantly. In the GSM8K, employing a single data augmentation method enables data reformulation to attain an accuracy rate of 59.6%. In the MATH, using only question alteration data yields a 17.9% accuracy rate. Surprisingly, when combining multiple data augmentation methods in any manner, each additional data increment contributes to further enhancement. This phenomenon persists even at high accuracy levels. This highlights the versatility and effectiveness of each augmentation method.

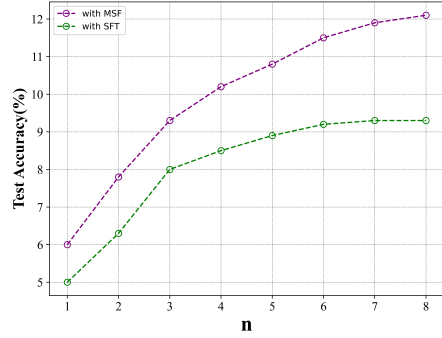


Figure 4: Comparison of performance between models trained with MSF and with SFT on MATH dataset.

### 4.2.3 MSF vs. SFT

We extract 7K new created questions from MATH to validate our proposed Majority Sampling Fine-tuning (MSF). Specifically, for each question we randomly select  $n$  solutions with the majority answer to construct MSF dataset (for those questions with less than  $n$  majority solutions, we compromise to use all the  $< n$  solutions), and directly request  $n$  solutions with different answers to construct SFT dataset. Figure 4 illustrates that as the amount of training data increases (with  $n$  varying from 1 to 8), models trained using MSF and SFT both see a progressive improvement in their performance. However, the latter saturates earlier than the former, and across all data sizes, the MSF models consistently outperform the SFT ones.

## 5 Conclusion

In this work, we propose four novel methods to broaden the scope of augmentation for mathematical reasoning data: solution reorganization, BF-Trans, expression replacement and nested multi-task construction. Through a variety of augmenting strategies, we create a multi-perspective mathematical problem-solving dataset based on GSM8K and MATH, called MuMath. After finetuning LLaMA-2 on the novel dataset, we get a series of models (7B, 13B and 70B) equipped with excellent math capability, which are also termed MuMath. Extensive empirical results demonstrate the effectiveness of our proposed augmentation methods. Compared to the open-source methods, our MuMath achieves the best performance in tool-free LLMs across all model scales, and even surpasses some tool-use counterparts. We will explore other augmentation methods for further improving mathematical reasoning performance of tool-free LLMs, as well as more auxiliary tasks for nested multi-task learning.



## References

- 543 Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.
- 548 Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. [Language models are few-shot learners](#).
- 559 Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. [Evaluating large language models trained on code](#).
- 579 Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2023. [Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks](#).
- 583 Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. [Palm: Scaling language modeling with pathways](#).
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yao Fu, Litu Ou, Mingyu Chen, Yuhao Wan, Hao Peng, and Tushar Khot. 2023a. [Chain-of-thought hub: A continuous effort to measure large language models’ reasoning performance](#).
- Yao Fu, Hao Peng, Litu Ou, Ashish Sabharwal, and Tushar Khot. 2023b. Specializing smaller language models towards multi-step reasoning. *arXiv preprint arXiv:2301.12726*.
- Yao Fu, Hao Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022. Complexity-based prompting for multi-step reasoning. *arXiv preprint arXiv:2210.00720*.
- Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. [Pal: Program-aided language models](#).
- Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujia Yang, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. [Tora: A tool-integrated reasoning agent for mathematical problem solving](#).
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021a. Measuring mathematical problem solving with the MATH dataset.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.
- Namgyu Ho, Laura Schmid, and Se-Young Yun. 2023. [Large language models are reasoning teachers](#).
- Jiaxin Huang, Shixiang Shane Gu, Le Hou, Yuexin Wu, Xuezhi Wang, Hongkun Yu, and Jiawei Han. 2022. [Large language models can self-improve](#).

654	Weisen Jiang, Han Shi, Longhui Yu, Zhengying Liu,	OpenAI. 2023a. Chatgpt: Optimizing language	711
655	Yu Zhang, Zhenguo Li, and James T. Kwok. 2023a.	models for dialogue. <a href="https://openai.com/blog/chatgpt">https://openai.com/blog/</a>	712
656	<a href="#">Forward-backward reasoning in large language mod-</a>	<a href="#">chatgpt</a> .	713
657	<a href="#">els for mathematical verification</a> .		
658	Weisen Jiang, Yu Zhang, and James Kwok. 2023b. Ef-	OpenAI. 2023b. <a href="#">Gpt-4 technical report</a> .	714
659	fective structured prompting by meta-learning and		
660	representative verbalizer. In <i>Proceedings of the 40th</i>	German I. Parisi, Ronald Kemker, Jose L. Part, Christo-	715
661	<i>International Conference on Machine Learning</i> , vol-	pher Kanan, and Stefan Wermter. 2019. <a href="#">Continual</a>	716
662	ume 202 of <i>Proceedings of Machine Learning Re-</i>	<a href="#">lifelong learning with neural networks: A review</a> .	717
663	<i>search</i> , pages 15186–15199. PMLR.	<i>Neural Networks</i> , 113:54–71.	718
664	Aitor Lewkowycz, Anders Andreassen, David Dohan,	A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, , and	719
665	Ethan Dyer, Henryk Michalewski, Vinay Ramasesh,	I. Sutskever. 2019. Language models are unsuper-	720
666	Ambrose Slone, Cem Anil, Imanol Schlag, Theo	vised multitask learners. <i>Technical Report</i> .	721
667	Gutman-Solo, et al. 2022. Solving quantitative rea-		
668	soning problems with language models. <i>Advances</i>	Colin Raffel, Noam Shazeer, Adam Roberts, Katherine	722
669	<i>in Neural Information Processing Systems</i> , 35:3843–	Lee, Sharan Narang, Michael Matena, Yanqi Zhou,	723
670	3857.	Wei Li, and Peter J. Liu. 2023. <a href="#">Exploring the limits</a>	724
671	Chengpeng Li, Zheng Yuan, Hongyi Yuan, Guanting	<a href="#">of transfer learning with a unified text-to-text trans-</a>	725
672	Dong, Keming Lu, Jiancan Wu, Chuanqi Tan, Xiang	<a href="#">former</a> .	726
673	Wang, and Chang Zhou. 2023. <a href="#">Query and response</a>		
674	<a href="#">augmentation cannot help out-of-domain math rea-</a>	Baptiste Rozière, Jonas Gehring, Fabian Gloeckle,	727
675	<a href="#">soning generalization</a> .	Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi	728
676	Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen,	Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, Artyom	729
677	Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian,	Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish	730
678	Baolin Peng, Yi Mao, Wenhui Chen, and Xifeng	Bhatt, Cristian Canton Ferrer, Aaron Grattafori, Wen-	731
679	Yan. 2022. <a href="#">Explanations from large language models</a>	han Xiong, Alexandre Défossez, Jade Copet, Faisal	732
680	<a href="#">make small reasoners better</a> .	Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier,	733
681	Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Man-	Thomas Scialom, and Gabriel Synnaeve. 2023. <a href="#">Code</a>	734
682	dar Joshi, Danqi Chen, Omer Levy, Mike Lewis,	<a href="#">llama: Open foundation models for code</a> .	735
683	Luke Zettlemoyer, and Veselin Stoyanov. 2019.		
684	<a href="#">Roberta: A robustly optimized bert pretraining ap-</a>	John Schulman, Filip Wolski, Prafulla Dhariwal, Alec	736
685	<a href="#">proach</a> .	Radford, and Oleg Klimov. 2017. <a href="#">Proximal policy</a>	737
686	Shayne Longpre, Le Hou, Tu Vu, Albert Webson,	<a href="#">optimization algorithms</a> .	738
687	Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V		
688	Le, Barret Zoph, Jason Wei, et al. 2023. The flan	Kumar Shridhar, Alessandro Stolfo, and Mrinmaya	739
689	collection: Designing data and methods for effective	Sachan. 2023. <a href="#">Distilling reasoning capabilities into</a>	740
690	instruction tuning. <i>arXiv preprint arXiv:2301.13688</i> .	<a href="#">smaller language models</a> . In <i>Findings of the Asso-</i>	741
691	Haipeng Luo, Qingfeng Sun, Can Xu, Pu Zhao, Jian-	<i>ciation for Computational Linguistics: ACL 2023</i> ,	742
692	guang Lou, Chongyang Tao, Xiubo Geng, Qingwei	pages 7059–7073, Toronto, Canada. Association for	743
693	Lin, Shifeng Chen, and Dongmei Zhang. 2023a. <a href="#">Wiz-</a>	Computational Linguistics.	744
694	<a href="#">ardmath: Empowering mathematical reasoning for</a>		
695	<a href="#">large language models via reinforced evol-instruct</a> .	Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao	745
696	Ziyang Luo, Can Xu, Pu Zhao, Qingfeng Sun, Xiubo	Tian, Hua Wu, and Haifeng Wang. 2019. <a href="#">Ernie 2.0:</a>	746
697	Geng, Wenxiang Hu, Chongyang Tao, Jing Ma, Qing-	<a href="#">A continual pre-training framework for language un-</a>	747
698	wei Lin, and Daxin Jiang. 2023b. Wizardcoder:	<a href="#">derstanding</a> .	748
699	Empowering code large language models with evol-		
700	instruct. <i>arXiv preprint arXiv:2306.08568</i> .	Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann	749
701	Lucie Charlotte Magister, Jonathan Mallinson, Jakub	Dubois, Xuechen Li, Carlos Guestrin, Percy Liang,	750
702	Adamek, Eric Malmi, and Aliaksei Severyn. 2023.	and Tatsunori B. Hashimoto. 2023. Stanford alpaca:	751
703	<a href="#">Teaching small language models to reason</a> .	An instruction-following llama model. <a href="https://github.com/tatsu-lab/stanford_alpaca">https://</a>	752
704	Sewon Min, Mike Lewis, Luke Zettlemoyer, and Han-	<a href="#">github.com/tatsu-lab/stanford_alpaca</a> .	753
705	naneh Hajishirzi. 2022. <a href="#">MetaICL: Learning to learn</a>		
706	<a href="#">in context</a> . In <i>Proceedings of the 2022 Conference of</i>	Hugo Touvron, Louis Martin, Kevin Stone, Peter Al-	754
707	<i>the North American Chapter of the Association for</i>	bert, Amjad Almahairi, Yasmine Babaei, Nikolay	755
708	<i>Computational Linguistics: Human Language Tech-</i>	Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti	756
709	<i>nologies</i> , pages 2791–2809, Seattle, United States.	Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton	757
710	Association for Computational Linguistics.	Ferrer, Moya Chen, Guillem Cucurull, David Esiobu,	758
		Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller,	759
		Cynthia Gao, Vedanuj Goswami, Naman Goyal, An-	760
		thony Hartshorn, Saghar Hosseini, Rui Hou, Hakan	761
		Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa,	762
		Isabel Kloumann, Artem Korenev, Punit Singh Koura,	763

764 Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Di-  
765 ana Liskovich, Yinghai Lu, Yuning Mao, Xavier Mar-  
766 tinet, Todor Mihaylov, Pushkar Mishra, Igor Moly-  
767 bog, Yixin Nie, Andrew Poulton, Jeremy Reizen-  
768 stein, Rashi Rungta, Kalyan Saladi, Alan Schelten,  
769 Ruan Silva, Eric Michael Smith, Ranjan Subrama-  
770 nian, Xiaoqing Ellen Tan, Binh Tang, Ross Tay-  
771 lor, Adina Williams, Jian Xiang Kuan, Puxin Xu,  
772 Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan,  
773 Melanie Kambadur, Sharan Narang, Aurelien Ro-  
774 driguez, Robert Stojnic, Sergey Edunov, and Thomas  
775 Scialom. 2023. [Llama 2: Open foundation and fine-  
776 tuned chat models.](#)

777 Ke Wang, Houxing Ren, Aojun Zhou, Zimu Lu, Sichun  
778 Luo, Weikang Shi, Renrui Zhang, Linqi Song,  
779 Mingjie Zhan, and Hongsheng Li. 2023. [Mathcoder:  
780 Seamless code integration in llms for enhanced math-  
781 ematical reasoning.](#)

782 Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten  
783 Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou,  
784 et al. 2022. Chain-of-thought prompting elicits rea-  
785 soning in large language models. *Advances in Neural  
786 Information Processing Systems*, 35:24824–24837.

787 Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He,  
788 Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao.  
789 2023. [Large language models are better reasoners  
790 with self-verification.](#) In *Findings of the Associa-  
791 tion for Computational Linguistics: EMNLP 2023*,  
792 pages 2550–2575, Singapore. Association for Com-  
793 putational Linguistics.

794 Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng,  
795 Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin  
796 Jiang. 2023. [Wizardlm: Empowering large language  
797 models to follow complex instructions.](#)

798 Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu,  
799 Zhengying Liu, Yu Zhang, James T. Kwok, Zhenguo  
800 Li, Adrian Weller, and Weiyang Liu. 2023. [Meta-  
801 math: Bootstrap your own mathematical questions  
802 for large language models.](#)

803 Zheng Yuan, Hongyi Yuan, Chengpeng Li, Guanting  
804 Dong, Keming Lu, Chuanqi Tan, Chang Zhou, and  
805 Jingren Zhou. 2023. [Scaling relationship on learning  
806 mathematical reasoning with large language models.](#)

807 Xiang Yue, Xingwei Qu, Ge Zhang, Yao Fu, Wenhao  
808 Huang, Huan Sun, Yu Su, and Wenhua Chen. 2023.  
809 [Mammoth: Building math generalist models through  
810 hybrid instruction tuning.](#)

## A Examples

### Example A.1: Reorganization

**Question:** Weng earns \$12 an hour for babysitting. Yesterday, she just did 50 minutes of babysitting. How much did she earn?

**Answer:(200 token length)**

1. Understand the Problem: - Weng is a babysitter who earns \$12 per hour. - Yesterday, she worked for 50 minutes. - Our task is to calculate how much she earned.

2. Calculate the Earning per Minute: - To calculate Weng's earning per minute, we divide her hourly earning by 60 (the total number of minutes in an hour):  $\$12 / 60 = \$0.2$  per minute.

3. Calculate Weng's Total Earnings: - We multiply Weng's earning per minute by the number of minutes she worked:  $\$0.2 \times 50 = \$10$ .

4. Answer: - Weng earned \$10 for her 50 minutes of babysitting.

**Answer:(520 token length)**

1. Analyze and Breakdown the Problem: - In this problem, Weng is earning a rate of \$12 per hour for babysitting. - However, she hasn't worked a full hour; she's only worked for 50 minutes. - Therefore, the challenge here is to figure out just how much she's earned for that 50 minutes, given that her hourly rate is \$12.

2. Convert Hours to Minutes: - Since the rate Weng is given is in hours but the time she worked is in minutes, we need to convert her hourly rate to a per minute rate to simplify the calculations. - There are 60 minutes in an hour, so we'll divide her hourly earnings by 60 to find out how much she earns per minute. - The calculation is as follows:  $12/60 = \$0.2$  per minute.

3. Connect to real-life situations: - Picture Weng jotting down her babysitting time and then calculating this on her calculator, or maybe she's keeping track in an app that monitors her babysitting earnings. - Knowing that she earns \$0.2 per minute, she would simply need to multiply this rate by the number of minutes she worked.

4. Calculate Earnings: - Now that we have Weng's per minute earning rate, we can multiply this by the number of minutes she actually worked to find out how much she earned. - Here's how that calculation looks: she worked for 50 minutes, so her earnings are computed as  $0.2 \times 50 = \$10$ .

5. Insert human element: - Now, Weng can feel confident that she's accurately calculated her earnings. She can spend or save this money as she chooses. Perhaps she's saving her earnings for a special purchase or maybe she counts on this income for regular expenses. Regardless, her diligence in accurately calculating her earnings assures her of what she can expect in her paycheck.

6. Answer and Conclusion: - After faithfully jotting down her babysitting time, carefully calculating her earnings, and then multiplying by the number of minutes she spent babysitting, Weng has found that she has earned \$10 for her 50 minutes of babysitting. The answer is: 10.

### Example A.2: The comparison of different alteration methods

**Question:** Randy has 60 mango trees on his farm. He also has 5 less than half as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

**Response:** Half of the number of Randy's mango trees is  $60 / 2 = 30$  trees. So Randy has  $30 - 5 = 25$  coconut trees. Therefore, Randy has  $60 + 25 = 85$  trees on his farm. The answer is: 85

**Change numbers:** Randy has 60 mango trees on his farm. He also has 10 more than twice as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

**Response:** Half of the number of Randy's mango trees is  $60 / 2 = 30$  trees. So Randy has  $30 - 10 = 20$  coconut trees. Therefore, Randy has  $60 + 20 = 80$  trees on his farm. The answer is: 80

**Expression replacement:** Randy has 60 mango trees on his farm. He also has 5 more than twice as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

**Response:** Twice the number of mango trees on Randy's farm is  $60 * 2 = 120$  trees. The total number of coconut trees on Randy's farm is 5 more than twice the number of mango trees, a total of  $120 + 5 = 125$  trees. Altogether, Randy has  $125 + 60 = 185$  trees on his farm. The answer is: 185

**Increase complexity:** Randy has 60 mango trees on his farm. If he plants 10 more mango trees and 8 more coconut trees, the ratio of the number of mango trees to coconut trees would be 21. How many trees does Randy have in all on his farm if he decides not to plant any additional trees?"

**Response:** If Randy plants 10 more mango trees and 8 more coconut trees, then the ratio of the number of mango trees to coconut trees would be 21. This gives:  $(m + 10) / (c + 8) = 21/1$  Simplify the Equation: - Distribute the left-hand side of the equation:  $m/c + (10/c) = 21$  - Substituting  $m=60$ , we get  $60/c + (10/c) = 21$  Solve for c: - Multiply both sides by c to eliminate the denominator:  $60 + 10 = 21c$  - Combine like terms  $70 = 21c$  - Solve for c:  $c = 70/21$   $c = 3.33$  (rounded off) Solve for m: - Substituting the value of c which is 3.33 and  $m=60$  into the equation  $(m + 10) / (c + 8) = 21/1$  to see if the ratio of the number of mango trees to coconut trees would be 21:  $(60+10) / (3.33+8) = 6.3636$  but 21/1 is not equal to 6.3636 -We conclude that planting 10 more mango trees and 8 more coconut trees are impossible, and Randy has 60 mango trees and no more. Answer: - Therefore, Randy has 60 trees altogether on his farm if he decides not to plant any additional trees. The answer is: 60

### Example A.3: Nested Multi-task Construction

**Question:**

A package of candy has 3 servings with 120 calories each. John eats half the package. How many calories did he eat?

**Original Solution:**

There were  $3 * 120 = 360$  calories in the package. So he ate  $360 / 2 = 180$  calories.

The answer is: 180.

**Nested Multi-task Solution:**

**[Outline]**

1. A package of candy has 3 servings.
2. Every serving contains 120 calories.
3. John eats half the package.
4. Our task is to calculate how many calories John did eat.

**[Plan]**

1. Calculate the total number of calories in the package.
2. Calculate the number of calories John did eat.

**[Execution]**

1. One serving of the candy contains 120 calories, and there are 3 servings, so the total number of calories in the package is  $3 * 120 = 360$  calories.

2. John eats half of the package, which contains  $360 / 2 = 180$  calories.

The answer is: 180.

## B The Prompts We Use

### Prompt B.1: Reorganization

You are a mathematics expert. Based on the provided questions and answer process, you reorganize the Solved Process to add the token length. Let's reorganize the Solved Process. During this process, You must expand the problem-solving process to approximately 420 tokens. The methods can include: 1. increasing the complexity of the problem-solving process; 2. adding extra concepts; 3. inserting a background story; 4. increasing the steps of solving the problem. The end of response needs to be: The answer is: [answer]. Here are two examples how to do it,

==== Example 1 =====

[Problem]:

Ralph is going to practice playing tennis with a tennis ball machine that shoots out tennis balls for Ralph to hit. He loads up the machine with 175 tennis balls to start with. Out of the first 100 balls, he manages to hit  $\frac{2}{5}$  of them. Of the next 75 tennis balls, he manages to hit  $\frac{1}{3}$  of them. Out of all the tennis balls, how many did Ralph not hit?

[Answers]:

Out of the first 100 balls, Ralph was able to hit  $\frac{2}{5}$  of them and not able to hit  $\frac{3}{5}$  of them,  $\frac{3}{5} \times 100 = 60$  tennis balls Ralph didn't hit. Out of the next 75 balls, Ralph was able to hit  $\frac{1}{3}$  of them and not able to hit  $\frac{2}{3}$  of them,  $\frac{2}{3} \times 75 = 50$  tennis balls that Ralph didn't hit. Combined, Ralph was not able to hit  $60 + 50 = 110$  tennis balls Ralph didn't hit. The answer is: 110

[Result]:

1. Understand the Problem:

- Here, Ralph is practicing with a tennis ball machine with 175 tennis balls.

- Ralph is hitting the balls in two batches, 100 and 75.

- From each batch, we know the fraction of balls Ralph was able to hit.

- Our task is to calculate how many tennis balls Ralph did not hit.

2. Identify the Fractions that Ralph doesn't hit:

- From the first 100 balls, Ralph manages to hit  $\frac{2}{5}$  of them, meaning he was not able to hit  $1 - \frac{2}{5}$  or  $\frac{3}{5}$  of these balls.

- Similarly, from the next 75 balls, Ralph manages to hit  $\frac{1}{3}$  of these, leaving  $1 - \frac{1}{3}$  or  $\frac{2}{3}$  of these balls that he doesn't hit.

3. Calculate the number of Missed Balls in the First Batch:

- We multiply the total number of balls in the first batch by the fraction that Ralph doesn't hit:

$\frac{3}{5} \times 100 = 60$  tennis balls Ralph didn't hit from the first batch.

4. Calculate the number of Missed Balls in the Second Batch:

- We multiply the total number of balls in the second batch by the fraction that Ralph doesn't hit:

$\frac{2}{3} \times 75 = 50$  tennis balls Ralph didn't hit from the second batch.

5. Calculate Total Missed Balls:

- Now that we have calculated the number of missed balls from both the first and second batch,

- Adding these two numbers together gives us the total missed balls:

$60 \text{ missed balls} + 50 \text{ missed balls} = 110 \text{ missed balls}$

6. Answer:

- So, out of all the tennis balls, Ralph didn't hit 110 tennis balls during his practice session.

The answer is: 110

==== Example 2 =====

[Problem]:

Mary is making ice cubes with fruit frozen in them for a cocktail party. She makes some strawberry cubes, and a number of blueberry cubes equal to 5 times the number of strawberry cubes minus 4. If she makes 116 ice cubes total, how many blueberry cubes does she make?

[Answers]: Let  $s$  be the number of strawberry cubes and  $b$  be the number of blueberry cubes. We know that  $b = 5s - 4$  and  $s + b = 116$ . Substituting the first equation into the second, we get  $s + 5s - 4 = 116$ . Combining like terms, we get  $6s - 4 = 116$ . Adding 4 to both sides, we get  $6s = 120$ . Dividing both sides by 6, we get  $s = 20$ . Substituting this value back into the equation  $b = 5s - 4$ , we get  $b = 96$ . The answer is: 96

[Result]:

1. Define the Variables:

- Let  $s$  represent the number of strawberry cubes.

- Let  $b$  represent the number of blueberry cubes.

2. Translate the Problem into Equations:

a. From the problem, we know that the number of blueberry cubes is 5 times the strawberry cubes minus 4. This gives:

$b = 5s - 4$

b. We also know the total number of cubes is 116. This gives:

$s + b = 116$

3. Substitute and Solve:

- Substitute the expression from the first equation into the second equation to eliminate  $b$ . - This gives:

$s + 5s - 4 = 116$

- Combine like terms:

$6s - 4 = 116$

- Solve for  $s$ :

Add 4 to both sides:

$6s = 120$

Divide both sides by 6:

$s = 20$

4. Find the Number of Blueberry Cubes:

- Using the value of  $s$  we found (20), plug it into the first equation to find  $b$ :

$b = 5(20) - 4$

- This gives:

$b = 96$

5. Answer: Mary makes 96 blueberry cubes for her cocktail party.

The answer is: 96

How about this question?

[Problem]:

[Answers]:

You must expand the problem-solving process to approximately 700 tokens. The end of response needs to be: The answer is: [answer].

[Result]:

## Prompt B.2: Prompt for BF-Trans GSM8K Questions

You are an experienced mathematics teacher in a grade school, and you are good at rephrase math problems.

Now you are given a math problem (marked as [Problem]) with one and only one  $X$  as the unknown variable. Your task is to rewrite or rephrase the original problem into an equivalent problem. The equivalent problem you rephrased should not contain any  $X$ s. Instead, you should ask for the correlated unknown value using a questioning tone in the last sentence of your rephrased problem. You can use more words to keep your rephrased problem expressed clearly and thoroughly, and also can add more concepts to avoid ambiguity. Here are some examples:

==== Example 1 ====

[Problem]:

Ralph is going to practice playing tennis with a tennis ball machine that shoots out tennis balls for Ralph to hit. He loads up the machine with 175 tennis balls to start with. Out of the first 100 balls, he manages to hit  $X$  of them. Of the next 75 tennis balls, he manages to hit  $1/3$  of them. Out of all the tennis balls, how many did Ralph not hit? If we know the answer to the above question is 110, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Ralph is going to practice playing tennis with a tennis ball machine that shoots out tennis balls for Ralph to hit. He loads up the machine with 175 tennis balls to start with, which are divided into 2 groups. In the first group there are 100 balls and the second group contains 75 ones. Of the second group of balls, Ralph manages to hit  $1/3$ . And out of all the tennis balls, Ralph did not hit 110. Then out of the first 100 balls, what is the proportion of the balls Ralph hit?

==== Example 2 ====

[Problem]:

In one day, 200 people visit The Metropolitan Museum of Art in New York City. Half of the visitors are residents of New York City. Of the NYC residents,  $X\%$  are college students. If the cost of a college student ticket is \$4, how much money does the museum get from college students that are residents of NYC? If we know the answer to the above question is 120, what is the value of the unknown variable  $X$ ?

[Rephrase]:

In one day, 200 people visit The Metropolitan Museum of Art in New York City. Half of the visitors are residents of New York City. If the cost of a college student ticket is \$4, and the museum gets \$120 from college students that are residents of NYC. Then of the NYC residents, what percentage is the college students?

==== Example 3 ====

[Problem]:

$X$  years from now, John will be 3 times as old as he was 11 years ago. How old is he now? If we know the answer to the above question is 21, what is the value of the unknown variable  $X$ ?

[Rephrase]:

If we know John is 21 years old, then how many years from now will John be 3 times as old as he was 11 years ago?

==== Example 4 ====

[Problem]:

Taipei 101 in Taiwan is  $X$  feet tall with 101 floors. Suppose the first to 100th floors have height each equal to 16.5 feet, how high is the 101st floor? If we know the answer to the above question is 23, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Taipei 101 in Taiwan has 101 floors. Suppose the first to 100th floors have height each equal to 16.5 feet, and the 101st floor is 23 feet. How high is the whole building?

==== Example 5 ====

[Problem]:

A fox can run at the maximum speed of  $X$  kilometers per hour. Considering the fox would run at a constant speed, what distance would he make during 120 minutes? If we know the answer to the above question is 100, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Considering a fox would run at a constant speed, and he will make 100 kilometers during 120 minutes. How many kilometers per hour the fox can run?

==== Example 6 ====

[Problem]:

Ruiz receives a monthly salary of \$500. If he received a  $X\%$  raise, how much will be Ruiz's new salary? If we know the answer to the above question is 530, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Ruiz receives a monthly salary of \$500. If his new salary will be \$530 monthly, what percentage is the raise?

==== Example 7 ====

[Problem]:

Tom decided to send his wife  $X$  dozen roses every day for the week. How many total roses did he send? If we know the answer to the above question is 168, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Tom sent his wife 168 roses totally for the week. How many dozen roses did he sent every day for the week?

==== Example 8 ====

[Problem]:

Facebook decided to award a productivity bonus to all its female employees who are mothers. This productivity bonus will total 25% of Facebook's annual earnings, which was  $X$  for the year 2020. It is known that Facebook employs 3300 employees; one-third are men, and of the women, 1200 are not mothers. How much was the bonus that each female mother employee received, assuming each one received an equal amount? If we know the answer to the above question is 1250, what is the value of the unknown variable  $X$ ?

[Rephrase]:

Facebook decided to award a productivity bonus to all its female employees who are mothers. This productivity bonus will total 25% of Facebook's annual earnings. It is known that Facebook employs 3300 employees; one-third are men, and of the women, 1200 are not mothers. Assuming each one received an equal amount, the bonus that each female mother employee received was \$1250. Then how much was the Facebook's annual earnings for the year?

==== Example 9 ====

[Problem]:

[Rephrase]:

### Prompt B.3: Request the answer to BF-Trans

You are an experienced mathematician. Now you are given a grade school math problem (marked as [Problem]). The task you should accomplish is to solve this problem.

You should solve the problem step by step, as thoroughly and clearly as you can by using any number of words or solution steps. Remember in the end of your solution you should output a "The answer is:" and then output the result answer number you get. Here are some examples:

==== Example 1 ====

[Problem]:

Ralph is going to practice playing tennis with a tennis ball machine that shoots out tennis balls for Ralph to hit. He loads up the machine with 175 tennis balls to start with, which are divided into 2 groups. In the first group there are 100 balls and the second group contains 75 ones. Of the second group of balls, Ralph manages to hit  $1/3$ . And out of all the tennis balls, Ralph did not hit 110. Then out of the first 100 balls, what is the proportion of the balls Ralph hit?

[Solution]:

Out of all the 175 tennis balls, Ralph did not hit 110, so he hit  $175 - 110 = 65$  of them. Of the second group of balls containing 75 balls, Ralph manages to hit  $1/3$ , so in this group he was able to hit  $75 \times 1/3 = 25$ . Of the first 100 balls, Ralph manages hit  $65 - 25 = 40$  tennis balls, so in this group the proportion of the balls Ralph hit is  $40 / 100 = 2/5$ . The answer is:  $2/5$

==== Example 2 ====

[Problem]:

In one day, 200 people visit The Metropolitan Museum of Art in New York City. Half of the visitors are residents of New York City. If the cost of a college student ticket is \$4, and the museum gets \$120 from college students that are residents of NYC. Then of the NYC residents, what percentage is the college students?

[Solution]:

200 people visit The Metropolitan Museum of Art in New York City, half of the visitors are residents of New York City, so the number of visitors that are residents of NYC is  $200 / 2 = 100$ . The cost of a college student ticket is \$4, and the museum gets \$120 from college students that are residents of NYC, so there are  $120 / 4 = 30$  college students that are residents of NYC. We know the number of visitors that are residents of NYC is 100, and there are 30 college students that are residents of NYC, so of the NYC residents, there is  $30 / 100 = 30\%$  college students, meaning the percentage is 30.

The answer is: 30

==== Example 3 ====

[Problem]:

Taipei 101 in Taiwan has 101 floors. Suppose the first to 100th floors have height each equal to 16.5 feet, and the 101st floor is 23 feet. How high is the whole building?

[Solution]:

The first to 100th floors have height each equal to 16.5 feet, so the total height of the first to 100th floors is  $16.5 \times 100 = 1650$  feet. We know the total height of the first to 100th floors is 1650 feet, and the 101st floor is 23 feet, so the whole building is  $1650 + 23 = 1673$  feet. The answer is: 1673

==== Example 4 ====

[Problem]:

Considering a fox would run at a constant speed, and he will make 100 kilometers during 120 minutes. How many kilometers per hour the fox can run?

[Solution]:

The fox will make 100 kilometers during 120 minutes, and 120 minutes are  $120 / 60 = 2$  hours, so he can run  $100 / 2 = 50$  kilometers per hour. The answer is: 50

==== Example 5 ====

[Problem]:

Facebook decided to award a productivity bonus to all its female employees who are mothers. This productivity bonus was total 25% of Facebook's annual earnings. It is known that Facebook employs 3300 employees; one-third are men, and of the women, 1200 are not mothers. Assuming each one received an equal amount, the bonus that each female mother employee received was \$1250. Then how much was the Facebook's annual earnings for the year?

[Solution]:

It is known that Facebook employs 3300 employees and  $1/3$  are men, so  $1 - 1/3 = 2/3$  are women and the number of women is  $3300 \times 2/3 = 2200$ .

Of the women, 1200 are not mothers, so there are  $2200 - 1200 = 1000$  mothers. Assuming each one received an equal amount, the productivity bonus that each female mother employee received was \$1250, and we know Of the women, there are 1000 mothers, so the total productivity bonus of the mother employees received was  $\$1250 \times 1000 = \$1,250,000$ .

We know the total productivity bonus of the mother employees received was \$1,250,000, and it's 25% of Facebook's annual earnings for the year, so Facebook's annual earnings for the year is  $\$1,250,000 / 25\% = \$1,250,000 / (1/4) = \$1,250,000 \times 4 = \$5,000,000$ . The answer is: 5,000,000

==== Example 6 ====

[Problem]:

[Solution]:



### Prompt B.4: Expression Replacement

You are a mathematics expert, and you need to help me rewrite a math problem. This math problem includes the question and an explanatory answer. First, you need to understand the question and explanation, then extract the arithmetic expression from the explanation in the question. Next, Then, randomly replace the arithmetic expressions, replace addition with subtraction, subtraction with addition, multiplication with division, and division with multiplication. You can randomly replace one or two operations. The key is to regenerate a corresponding question based on the replaced arithmetic expression while ensuring that it makes sense logically. Follow the given examples:

==== Example 1 ====

[Question]:

Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?

[Response]:

Natalia sold  $48/2 = \langle 48/2=24 \rangle 24$  clips in May. Natalia sold  $48+24 = \langle 48+24=72 \rangle 72$  clips altogether in April and May. The answer is: 72

[Mathematical expression]:

$48/2 = \langle 48/2=24 \rangle 24$ ,  $48+24 = \langle 48+24=72 \rangle 72$

[Changed mathematical expression]:

$48*2 = \langle 48*2=96 \rangle 96$ ,  $48+96 = \langle 48+96=144 \rangle 144$

[Changed Question]:

Natalia sold clips to 48 of her friends in April, and then she sold double as many clips in May. How many clips did Natalia sell altogether in April and May?

==== Example 2 ====

[Question]:

Bella bought stamps at the post office. Some of the stamps had a snowflake design, some had a truck design, and some had a rose design. Bella bought 15 snowflake stamps. She bought 9 more truck stamps than snowflake stamps, and 3 fewer rose stamps than truck stamps. How many stamps did Bella buy in all?

[Response]:

The number of truck stamps is  $15 + 9 = \langle 15+9=24 \rangle 24$ . The number of rose stamps is  $24 - 3 = \langle 24-3=21 \rangle 21$ . Bella bought  $15 + 24 + 21 = \langle 15+24+21=60 \rangle 60$  stamps in all. The answer is: 60

[Mathematical expression]:

$15 + 9 = \langle 15+9=24 \rangle 24$ ,  $24 - 3 = \langle 24-3=21 \rangle 21$ ,  $15 + 24 + 21 = \langle 15+24+21=60 \rangle 60$

[Changed mathematical expression]:

$15 - 9 = \langle 15-9=6 \rangle 6$ ,  $6 - 3 = \langle 6-3=3 \rangle 3$ ,  $15 + 6 + 3 = \langle 15+6+3=24 \rangle 24$

[Changed Question]:

Bella bought stamps at the post office. Some of the stamps had a snowflake design, some had a truck design, and some had a rose design. Bella bought 15 snowflake stamps. She bought 9 less truck stamps than snowflake stamps, and 3 fewer rose stamps than truck stamps. How many stamps did Bella buy in all?

==== Example 3 ====

[Question]:

Randy has 60 mango trees on his farm. He also has 5 less than half as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

[Response]:

Half of the number of Randy's mango trees is  $60/2 = \langle 60/2=30 \rangle 30$  trees. So Randy has  $30 - 5 = \langle 30-5=25 \rangle 25$  coconut trees. Therefore, Randy has  $60 + 25 = \langle 60+25=85 \rangle 85$  trees on his farm. The answer is: 85

[Mathematical expression]:

$60/2 = \langle 60/2=30 \rangle 30$ ,  $30 - 5 = \langle 30-5=25 \rangle 25$ ,  $60 + 25 = \langle 60+25=85 \rangle 85$

[Changed mathematical expression]:

$60/2 = \langle 60/2=30 \rangle 30$ ,  $30 + 5 = \langle 30+5=35 \rangle 35$ ,  $60 + 35 = \langle 60+35=95 \rangle 95$

[Changed Question]:

Randy has 60 mango trees on his farm. He also has 5 more than half as many coconut trees as mango trees. How many trees does Randy have in all on his farm?

How about this question?

[Question]:

[Response]:

### Prompt B.5: Request the answer to expression replacement questions

I want you to act as an excellent math solver. You will solve the given math question step by step. Retain decimals to three decimal places. The formulas in the process need to use the format:  $48/2 = \langle 48/2=24 \rangle 24$  clips. The end of response needs to be: The answer is: [answer]. Most importantly, if something doesn't make sense in the question, just write out: Sorry, this question is wrong. Follow the given examples:

==== Example 1 ====

[Question]:

Studying for her test, Mitchell had read ten chapters of a book before 4 o'clock. When it clocked 4, Mitchell had read 20 pages of the 11th chapter of the book she was studying from. After 4 o'clock, she didn't read the remaining pages of chapter eleven but proceeded and read 2 more chapters of the book. If each chapter in the book had 40 pages, calculate the total number of pages that Mitchell had read altogether?

[Result]:

Since each chapter of the book has 40 pages, Mitchell had read  $10 * 40 = \langle 10 * 40 = 400 \rangle 400$  pages from the first ten chapters. After reading 20 pages of the eleventh chapter, the total number of pages that Mitchell had read is  $400 + 20 = \langle 400 + 20 = 420 \rangle 420$ . The next two chapters that she read had  $2 * 40 = \langle 2 * 40 = 80 \rangle 80$  pages. In total, Mitchell read  $420 + 80 = \langle 420 + 80 = 500 \rangle 500$  pages of the book that day. The answer is: 500

==== Example 2 ====

[Question]:

Fern is checking IDs to get into an R-rated movie. She denied 20% of the 120 kids from Riverside High, 70% of the 90 kids from West Side High, and half the 50 kids from Mountaintop High. How many kids got into the movie?

[Result]:

First find how many kids from Riverside High are rejected:  $20\% * 120 \text{ kids} = \langle 20 * .01 * 120 = 24 \rangle 24$  kids. Then find how many kids from West Side High are rejected:  $70\% * 90 \text{ kids} = \langle 70 * .01 * 90 = 63 \rangle 63$  kids. Then find how many kids from Mountaintop High are rejected:  $50 \text{ kids} / 2 = \langle 50 / 2 = 25 \rangle 25$  kids. Then add the number of kids from each school to find the total number of kids:  $120 \text{ kids} + 90 \text{ kids} + 50 \text{ kids} = \langle 120 + 90 + 50 = 260 \rangle 260$  kids. Then subtract all the kids who were rejected from the total number of kids to find the number who got in:  $260 \text{ kids} - 24 \text{ kids} - 63 \text{ kids} - 25 \text{ kids} = \langle 260 - 24 - 63 - 25 = 148 \rangle 148$  kids. The answer is: 148

==== Example 3 ====

[Question]:

After tests in California, the total number of Coronavirus cases was recorded as 2000 positive cases on a particular day. The number of cases increased by 500 on the second day, with 50 recoveries. On the third day, the total number of new cases spiked to 1500 with 200 recoveries. What's the total number of positive cases after the third day?

[Result]:

When 500 new cases were recorded after the tests, the total number of positive cases increased to  $2000 \text{ cases} + 500 \text{ cases} = \langle 2000 + 500 = 2500 \rangle 2500$  cases. With 50 recoveries, the total number of cases reduced to  $2500 \text{ cases} - 50 \text{ cases} = \langle 2500 - 50 = 2450 \rangle 2450$  cases. On the third day, with 1500 new cases, the total number of cases became  $2450 \text{ cases} + 1500 \text{ cases} = \langle 2450 + 1500 = 3950 \rangle 3950$  cases. If 200 people recovered from the virus, the total number of people with Coronavirus became  $3950 \text{ cases} - 200 \text{ cases} = 3750 \text{ cases}$ . The answer is: 3750"

==== Example 4 ====

[Question]:

Lisa and Carly go shopping together. Lisa spends \$40 on t-shirts then spends half of this amount on jeans and twice this amount on coats. Carly spends only a quarter as much as Lisa on t-shirts but spends 3 times as much on jeans and a quarter of the amount Lisa spent on coats. In dollars, how much did Lisa and Carly spend in total?

[Result]:

Lisa spends \$40 on t-shirts / 2 =  $\langle 40 / 2 = 20 \rangle 20$  on jeans. She also spends  $\$40 \text{ on t-shirts} * 2 = \langle 40 * 2 = 80 \rangle 80$  on coats. So Lisa has spent a total of  $40 + 20 + 80 = \langle 40 + 20 + 80 = 140 \rangle 140$ . Carly spends  $\$40 / 4 = \langle 40 / 4 = 10 \rangle 10$  on t-shirts. She also spends  $\$20 \text{ per pair of jeans} * 3 = \langle 20 * 3 = 60 \rangle 60$  on jeans. She then also spends  $\$80 \text{ Lisa's cost for coats} / 4 = \langle 80 / 4 = 20 \rangle 20$  on coats. So Carly has spent a total of  $10 + 60 + 20 = \langle 10 + 60 + 20 = 90 \rangle 90$ . Lisa and Carly have therefore spent a total of  $140 + 90 = \langle 140 + 90 = 230 \rangle 230$ . The answer is: 230"

==== Example 5 ====

[Question]:

In a section of the forest, there are 100 weasels and 50 rabbits. Three foxes invade this region and hunt the rodents. Each fox catches an average of 4 weasels and 2 rabbits per week. How many rabbits and weasels will be left after 3 weeks?

[Result]:

3 foxes catch 4 weasels each every week for a total of  $3 * 4 = \langle 3 * 4 = 12 \rangle 12$  weasels. 12 weasels are caught every week for 3 weeks for a total of  $12 * 3 = \langle 12 * 3 = 36 \rangle 36$  weasels. 3 foxes catch 2 rabbits each every week for a total of  $3 * 2 = \langle 3 * 2 = 6 \rangle 6$  rabbits. 6 rabbits are caught every week for 3 weeks for a total of  $6 * 3 = \langle 6 * 3 = 18 \rangle 18$  rabbits. There were originally 100 weasels so now there are  $100 - 36 = \langle 100 - 36 = 64 \rangle 64$  weasels left. There were originally 50 rabbits so now there are  $50 - 18 = \langle 50 - 18 = 32 \rangle 32$  rabbits left. There are  $64 + 32 = \langle 64 + 32 = 96 \rangle 96$  weasels and rabbits left. The answer is: 96"

[Question]:

[Result]:

## Prompt B.6: Nested Multi-task Learning

You are an experienced mathematics teacher in a grade school. Now you are given a grade school problem marked as [Problem] and its correlated solution marked as [Solution]. In the end of the [Solution], there is always a certain number after a "The answer is: " as the result answer. Based on the [Problem] and the corresponding [Solution], You are asked to generate a new solution, which is much clearer than the original one and much easier to understand even for the worst student.

The new solution you generate must by order contains [Outline], [Plan] and [Execution]. The [Outline] is an outline or summary of the [Problem]; the [Plan] is a plan as an ordered list of steps solving the problem; the [Execution] is an ordered list of your specific and detailed solving steps, each of which should be as thorough and clear as possible. There is a one-to-one correspondence between [Plan] list and [Execution] list. To make your new solution helpful and easy to understand, you may: 1, increase the number of solving steps in [Plan] and [Execution] lists; 2, explain with more words in each step of [Execution] list; 3, use several substeps in one step and even use subsubsteps in one substep for [Execution] list; 4, calculate each formula or equation step by step. In the end of the [Execution], you should also output a "The answer is: " and then output a number as your result answer. To guarantee the correctness of the result answer in the new solution you will generate, you are recommended to refer to the original [Solution] or even follow it. Now every time when you see "[Generate]:", you should start to generate your new solution. Here are some examples:

==== Example 1 =====

[Problem]:

Ralph is going to practice playing tennis with a tennis ball machine that shoots out tennis balls for Ralph to hit. He loads up the machine with 175 tennis balls to start with. Out of the first 100 balls, he manages to hit  $\frac{2}{5}$  of them. Of the next 75 tennis balls, he manages to hit  $\frac{1}{3}$  of them. Out of all the tennis balls, how many did Ralph not hit?

[Solution]:

Out of the first 100 balls, Ralph was able to hit  $\frac{2}{5}$  of them and not able to hit  $\frac{3}{5}$  of them,  $\frac{3}{5} \times 100 = 60$  tennis balls Ralph didn't hit. Out of the next 75 balls, Ralph was able to hit  $\frac{1}{3}$  of them and not able to hit  $\frac{2}{3}$  of them,  $\frac{2}{3} \times 75 = 50$  tennis balls that Ralph didn't hit. Combined, Ralph was not able to hit  $60 + 50 = 110$  tennis balls Ralph didn't hit. The answer is: 110

[Generate]:

[Outline]

Ralph loads up the machine with 175 balls totally.

Ralph practices with 2 group of balls, the first group contains 100 balls and the second group contains 75 balls.

Ralph manages to hit  $\frac{2}{5}$  balls of the first group, and manages to hit  $\frac{1}{3}$  balls of the second group.

Our task is to calculate how many tennis balls Ralph did not hit.

[Plan]

1. Calculate the proportion of balls that Ralph did not hit for the first group.
2. Calculate the number of balls that Ralph did not hit for the first group.
3. Calculate the proportion of balls that Ralph did not hit for the second group.
4. Calculate the number of balls that Ralph did not hit for the second group.
5. Calculate the total number of balls that Ralph did not hit for the first and the second group.

[Execution]

1. Calculate the proportion of balls that Ralph did not hit for the first group:

From the first group of 100 balls, Ralph manages to hit  $\frac{2}{5}$  of them, so he was not able to hit  $1 - \frac{2}{5} = \frac{3}{5}$  of these balls.

2. Calculate the number of balls that Ralph did not hit for the first group:

The first group contains 100 balls, the proportion of balls that Ralph did not hit for the first group is  $\frac{3}{5}$ , so the number of balls that Ralph did not hit for the first group is  $\frac{3}{5} \times 100 = 60$  balls.

3. Calculate the proportion of balls that Ralph did not hit for the second group:

From the second group of 75 balls, Ralph manages to hit  $\frac{1}{3}$  of them, so he was not able to hit  $1 - \frac{1}{3} = \frac{2}{3}$  of these balls.

4. Calculate the number of balls that Ralph did not hit for the second group:

The second group contains 75 balls, the proportion of balls that Ralph did not hit for the second group is  $\frac{2}{3}$ , so the number of balls that Ralph did not hit for the second group is  $\frac{2}{3} \times 75 = 50$  balls.

5. Calculate the total number of balls that Ralph did not hit for the first and the second group:

The number of balls that Ralph did not hit for the first group is 60, the number of balls that Ralph did not hit for the second group is 50, so the total number of balls that Ralph did not hit for the first and the second group is  $60 + 50 = 110$ .

The answer is: 110

==== Example 2 =====

[Problem]:

In one day, 200 people visit The Metropolitan Museum of Art in New York City. Half of the visitors are residents of New York City. Of the NYC residents, 30% are college students. If the cost of a college student ticket is \$4, how much money does the museum get from college students that are residents of NYC?

[Solution]:

The number of visitors that are residents of NYC is  $200 / 2 = 100$  visitors The number of NYC resident visitors that are college students is  $100 * 0.30 = 30$  visitors The money from the college students that are residents of NYC is  $30 * 4 = 120$  The answer is: 120

[Generate]:

[Outline]

200 people visit the museum.

Half of the visitors are residents of NYC.

Of the NYC residents, 30% are college students.

The cost of a college student ticket is \$4.

Our task is to calculate how much money the museum gets from college students that are residents of NYC.

[Plan]

1. Calculate the number of visitors that are residents of NYC.
2. Calculate the number of NYC resident visitors that are college students.
3. Calculate the money from the college students that are residents of NYC.

[Execution]

1. Calculate the number of visitors that are residents of NYC:

200 people visit the museum totally, half of the visitors are residents of NYC, so the number of visitors that are residents of NYC is  $200 / 2 = 100$  visitors.

2. Calculate the number of NYC resident visitors that are college students:

The number of visitors that are residents of NYC is 100, and of them 30% are college students, so the number of NYC resident visitors that are college students is  $100 * 0.30 = 30$  visitors.

3. Calculate the money from the college students that are residents of NYC:

The number of NYC resident visitors that are college students is 30, and the cost of a college student ticket is \$4, so the money from the college students that are residents of NYC is  $30 * 4 = 120$

The answer is: 120

==== Example 3 =====

[Problem]:

[Solution]:

[Generate]: