# SCIBENCH: Evaluating College-Level Scientific Problem-Solving Abilities of Large Language Models

Xiaoxuan Wang [* 1]  Ziniu Hu [* 2]  Pan Lu [* 1]  Yanqiao Zhu [* 1]  Jieyu Zhang [3]  Satyen Subramaniam [1]
Arjun R. Loomba [1]  Shichang Zhang [1]  Yizhou Sun [1]  Wei Wang [1]

Project Homepage: `https://scibench-ucla.github.io`

## Abstract

Most existing Large Language Model (LLM) benchmarks on scientific problem reasoning focus on problems grounded in high-school subjects and are confined to elementary algebraic operations. To systematically examine the reasoning capabilities required for solving complex scientific problems, we introduce an expansive benchmark suite SCIBENCH for LLMs. SCIBENCH contains a carefully curated dataset featuring a range of collegiate-level scientific problems from mathematics, chemistry, and physics domains. Based on the dataset, we conduct an in-depth benchmarking study of representative open-source and proprietary LLMs with various prompting strategies. The results reveal that current LLMs fall short of delivering satisfactory performance, with the best overall score of merely 43.22%. Furthermore, through a detailed user study, we categorize the errors made by LLMs into ten problem-solving abilities. Our analysis indicates that no single prompting strategy significantly outperforms the others and some strategies that demonstrate improvements in certain problem-solving skills could result in declines in other skills. We envision that SCIBENCH will catalyze further developments in the reasoning abilities of LLMs, thereby ultimately contributing to scientific research and discovery.

## 1. Introduction

Recent advancements in Large Language Models (LLMs) have dramatically expanded the boundaries of artificial intelligence (Brown et al., 2020; Gao et al., 2023; Liu et al., 2023b; OpenAI., 2022; Touvron et al., 2023a; Zhang et al., 2023a;b). They have demonstrated outstanding performance in many mathematical reasoning tasks that are typically considered challenging even for well-educated individuals (Chen et al., 2021; 2023a; Gao et al., 2022; Kojima et al., 2022; Wei et al., 2022). Notably, GPT-4 achieves a remarkable score of 163 out of 170 on the GRE Quantitative Exam, placing it at the 80th percentile ranking (OpenAI., 2023).

While the remarkable improvements in these benchmark performances might suggest that LLMs are capable of performing scientific reasoning tasks, we argue that this assertion might be overly optimistic due to the inherent limitations of current benchmarks. Firstly, many existing benchmarks such as ScienceQA (Lu et al., 2022) and GSM8K (Cobbe et al., 2021) only contain problems grounded in grade-level subjects. Although other benchmarks like MATH (Hendrycks et al., 2021) introduce high-school level questions, they primarily focus on math problems. Secondly, recent works like MMLU (Hendrycks et al., 2020), AGIEval (Zhong et al., 2023), and JEEBench (Arora et al., 2023), despite introducing challenging problems that span a wide range of disciplines, only require basic computations—addition, subtraction, multiplication, and exponentiation—which do not adequately assess the depth of reasoning abilities of LLMs for solving scientific problems. Lastly, most of these benchmarks only include textual problems, which omit problems that incorporate visual elements such as figures or diagrams.

In parallel to benchmark developments, many studies propose various prompting strategies aimed at enhancing the reasoning abilities of LLMs in scientific problem solving. A notable example is the Chain-of-Thought (CoT) approach, which instructs LLMs to generate detailed, step-by-step solutions that prompt deeper problem thinking (Huang et al., 2022; Wang et al., 2022; Wei et al., 2022; Zhou et al., 2022). Additionally, other strategies propose to enable LLMs to utilize external tools (Lu et al., 2023b; Schick et al., 2023) that improve the numerical computation capability. However, even these strategic approaches, each with its specific strengths, struggle to fully address complex scientific prob-

---

[*]Equal contribution [1]University of California, Los Angeles, Los Angeles, CA, USA [2]California Institute of Technology, Pasadena, CA, USA [3]University of Washington, Seattle, WA, USA. Correspondence to: Xiaoxuan Wang <xw27@cs.ucla.edu>.

**Problem (Using the Planck distribution)**
Compare the energy output of a black-body radiator (such as an incandescent lamp) at two different wavelengths by calculating the ratio of the energy output at 450 nm (blue light) to that at 700 nm (red light) at 298 K.

**Correct Solution**
At a temperature $T$, the ratio of the spectral density of states at a wavelength $\lambda_1$ to that at $\lambda_2$ is given by

$$\frac{\rho(\lambda_1, T)}{\rho(\lambda_2, T)} = \left(\frac{\lambda_2}{\lambda_1}\right)^5 \times \frac{(e^{hc/\lambda_2 kT} - 1)}{(e^{hc/\lambda_1 kT} - 1)}$$

Insert the data and evaluate this ratio.
Answer With $\lambda_1 = 450$ nm and $\lambda_2 = 700$ nm, ..., and therefore

$$\frac{\rho(450 \text{ nm}, 298 \text{ K})}{\rho(700 \text{ nm}, 298 \text{ K})} = 2.10 \times 10^{-16}$$

**Final Answer:** $2.10 \times 10^{-16}$

**GPT Solution with CoT Prompting**
**Step 1:** We need to use the Planck distribution formula to calculate the energy output of a black-body radiator. The formula is:
$$B(\lambda, T) = \frac{2hc^2}{\lambda^5} \frac{1}{e^{\frac{hc}{\lambda k_B T}} - 1}$$
**Step 2:** We calculate the energy output at 450 nm and 700 nm... Let's use $T = 298$ K.
$$B(450 \text{ nm}, 298 \text{ K})$$
$$= \frac{2hc^2}{(450 \times 10^{-9} \text{ m})^5} \frac{1}{e^{\frac{hc}{450 \times 10^{-9} \text{ m} \cdot k_B \cdot 298 \text{ K}}} - 1}$$
$$\approx 1.23 \times 10^{-16} \quad \Rightarrow 1.67 \times 10^{-31}$$
...
**Step 3:** Calculate the ratio of the energy output at 450 nm to that at 700 nm:
$$\frac{B(450 \text{ nm}, 298 \text{ K})}{B(700 \text{ nm}, 298 \text{ K})} \approx 4.39$$

**Final Answer:** $4.39 \times 10^{-16}$

**GPT Solution with Python Prompting as External Tools**

... The ratio of the energy output at $\lambda_1 = 450$ nm and $\lambda_2 = 700$ nm is
$$\frac{u(\lambda_1, T)}{u(\lambda_2, T)} = \frac{\frac{8\pi hc}{\lambda_1^5} \frac{1}{e^{hc/(\lambda_1 kT)} - 1}}{\frac{8\pi hc}{\lambda_2^5} \frac{1}{e^{hc/(\lambda_2 kT)} - 1}} = \frac{\lambda_2^5}{\lambda_1^5} \frac{e^{hc/(\lambda_1 kT)} - 1}{e^{hc/(\lambda_2 kT)} - 1}$$
$\Rightarrow \lambda_2$ (over $\lambda_1$ in numerator), $\Rightarrow \lambda_1$ (over $\lambda_2$ in denominator)

Substituting the given values, we get a Python code for this problem:
```
…                              ⇒lambda2
ratio = ((lambda2 / lambda1)**5)
*((math.exp((h * c) / (lambda1 * k *
T)) - 1) / (math.exp((h * c) /
(lambda2 * k * T)) - 1))
⇒lambda1
…
```

**Final Answer:** $3.95 \times 10^{33}$

*Figure 1.* An example problem from *Physical Chemistry* with solutions generated under two prompting strategies. GPT-4 with Chain-of-Thought (CoT) prompting shows calculation errors, while GPT-4 that prompts Python as external tools misunderstands mathematical equations. Errors are highlighted in red and the corrections are shown in purple.

lems. Consider an example problem from college-level *Physical Chemistry* (Atkins et al., 2014b) that requires the use of the Planck distribution to derive certain quantities. As shown in Figure 1, LLMs with CoT prompts accurately generate the correct formula, but fail in the final numerical calculation. As a remedy, when instructed to simultaneously generate a Python program for numerical computation and employ the CoT reasoning, the LLM misplaces $\lambda_1$ in the numerator rather than the denominator in the formula, illustrating a misunderstanding of mathematical relationships when employing external tools. This example highlights a crucial gap: even advanced LLMs struggle with complex scientific problem solving, necessitating a fine-grained analysis of the skills required for such complex tasks.

To mitigate these deficiencies, in this paper, we present a novel college-level **Sci**entific problem solving **Bench**mark, referred to as SciBench. SciBench contains a carefully curated dataset of college-level scientific problems, including 869 problems collected from widely-used textbooks in college-level Chemistry, Physics, and Mathematics courses. Distinct from existing benchmarks, all of the problems are open-ended, free-response questions that demand multi-step reasoning abilities, the understanding of scientific concepts, the retrieval of domain-specific knowledge (e.g., equations and theorems), and complex numeric computation capabilities (e.g., calculus or differential equations). Besides that, our dataset includes a multimodal subset of 177 problems that incorporate visual elements (such as graphs and figures) as additional contexts, which enables of the evaluation of multimodal LLMs. It is noted that SciBench also includes step-by-step solutions for example problems, facilitating detailed error analysis. To align our evaluation with real-

world scenarios, we provide a separate, closed dataset that encompasses 103 problems from seven sets of midterm and final exams from collegiate Computer Science and Math courses. To ensure the integrity of our evaluation, these datasets have been manually extracted from PDF documents and formatted into LaTeX documents, thereby minimizing the risk of their leakage in LLM training data.

Our evaluation includes a wide range of representative open-source and proprietary LLMs. For unimodal, textual-based LLMs, we assess LLaMA-2, Mistral, Claude2, GPT-3.5, GPT-4, and their variants. For multimodal vision-language models, we include GPT-4, InternLM-XComposer2, Qwen-VL, SPHINX-MoE, and LLaVA. These models are tested using various prompting strategies, including CoT, zero-shot learning, and few-shot learning. We also prompt LLMs to utilize external scientific computing libraries in Python and Wolfram language. The experimental results indicate that the complexity and difficulty of our dataset are sufficient to differentiate the performance levels of different LLMs. Even with the strongest configuration—combining CoT prompting and the use of external tools—the best model achieves an average score of 43.22% on the textual dataset, 13.8% on the multimodal dataset, and 51.57% on the closed exam dataset. These results suggest a considerable potential for improvement in future LLMs.

In order to gain a comprehensive understanding of the limitations of LLMs in scientific problem solving, we propose a novel self-refinement method to uncover the deficient skills in the solutions made by LLMs. Firstly, we compare the correct solutions with the solutions generated by LLMs and, with the assistance of human annotators, summarize ten essential skills requisite for successful scientific

*Table 1.* Comparison of SCIBENCH with other benchmarks. "Algebra" refers to high-school level arithmetic computations; "Calculus" involves using integrals and differentials; "Statistics" focuses on applying statistical and probability concepts like bivariate distributions.

| Benchmark | Subject | | | Calculation | | | College Level | Visual Contexts | Detailed Solutions | Free Response |
|---|---|---|---|---|---|---|---|---|---|---|
| | Math | Chemistry | Physics | Algebra | Calculus | Statistics | | | | |
| ScienceQA (Lu et al., 2022) | ✓ | ✓ | ✓ | ✓ | | | | ✓ | | |
| IconQA (Lu et al., 2021b) | ✓ | ✓ | ✓ | ✓ | | | | ✓ | | |
| TabMWP (Lu et al., 2023c) | ✓ | | | ✓ | | | | | ✓ | ✓ |
| GSM8K (Cobbe et al., 2021) | ✓ | | | ✓ | | | | | ✓ | ✓ |
| MATH (Hendrycks et al., 2021) | ✓ | | | ✓ | | | | | ✓ | ✓ |
| LILA (Mishra et al., 2022) | ✓ | | | ✓ | | | | | ✓ | ✓ |
| MMLU (Hendrycks et al., 2020) | ✓ | | | ✓ | | | ✓ | | ✓ | |
| TheroemQA (Chen et al., 2023b) | ✓ | | | ✓ | ✓ | | ✓ | | ✓ | ✓ |
| AGIEval (Zhong et al., 2023) | ✓ | ✓ | ✓ | ✓ | | | ✓ | | | |
| SciEval (Sun et al., 2023) | ✓ | ✓ | ✓ | ✓ | | | | | | |
| JEEBench (Arora et al., 2023) | ✓ | ✓ | ✓ | ✓ | | | | | ✓ | ✓ |
| SCIBENCH | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

problem-solving. These skills include proficiency in domain knowledge, mathematical reasoning, numerical calculation abilities, and comprehension of common sense concepts. Subsequently, we employ an LLM-empowered self-critic approach to automatically classify the lacking skills in the solutions made by the benchmarked LLMs under each experiment configuration. Our analysis finds that (1) although CoT significantly improves the calculation ability, it is less effective in other aspects; (2) prompts with the use of external tools could potentially compromise other fundamental skills; (3) few-shot learning does not universally improve scientific problem-solving skills.

## 2. Related Work

Recently, many benchmarks have been proposed to assess the scientific problem-solving skills of LLMs, particularly in mathematical domains (Chen et al., 2023b; Fu et al., 2023; Guo et al., 2023; Hendrycks et al., 2020; Lu et al., 2023c;d; Mishra et al., 2022; Welleck et al., 2021; Zhong et al., 2023). Notable works include GSM8K (Cobbe et al., 2021) including 8.5K grade school math word problems; LILA (Mishra et al., 2022) which extends 20 datasets with task instructions and Python solutions; MATH (Hendrycks et al., 2021), a challenging collection of 12.5K math problems from math competitions; TheroemQA (Chen et al., 2023b), focusing on theorem applications on problem solving; and MathVista (Lu et al., 2023a), which evaluates the mathematical reasoning ability of LLMs in visual contexts.

To provide a more holistic evaluation, recent studies have expanded their scope to multiple disciplines: ScienceQA (Lu et al., 2022) introduces a multimodal question-answering dataset with accompanying lecture notes and explanatory annotations. Taylor et al. (2022) provide a set of scientific tasks, including LaTeX equation conversions, domain knowledge probes, citation prediction, and chemical question answering. BIG-Bench (Ghazal et al., 2013) offers a large-scale general-purpose test suite that requires 204 multiple-choice or exact-match tasks, and its extension BIG-Bench Hard (Suzgun et al., 2022) poses challenging CoT prompts. SciEval (Sun et al., 2023) includes a mix of objective and subjective questions across multiple scientific fields to assess understanding, application, and research capabilities. JEEBench (Arora et al., 2023) incorporates pre-engineering-level scientific problems derived from college entrance exams. AGIEval (Zhong et al., 2023) evaluates LLMs on human-centric standardized exams, such as college entrance exams and lawyer qualification tests.

Despite their extensive coverage across diverse disciplines, these datasets exhibit certain limitations. Sourced from lower educational level subjects, the majority of them focus on basic arithmetic operations rather than advanced mathematical computations. Furthermore, most of these benchmarks are confined to textual-only problems, omitting problems with visual elements such as graphs or diagrams. These drawbacks result in an incomplete assessment of the analytical and problem-solving skills required to tackle complex scientific problems. In contrast, SCIBENCH focuses on college-level scientific problems across a broad spectrum of disciplines including Mathematics, Physics, and Chemistry. It emphasizes on a deep understanding of diverse scientific concepts, challenging LLMs to not only grasp these principles but also to efficiently retrieve and apply relevant knowledge. Furthermore, it demands sophisticated numerical computation skills, including the execution of advanced mathematical operations such as calculus and differential equations, as well as the application of advanced statistical and probability theories. Additionally, we include multimodal problems that necessitate the interpretation and integration of both textual and visual information. A detailed comparison of SCIBENCH with some representative works is summarized in Table 1.

*Table 2.* Summary of the textbook dataset. We report the number of total problems, percentage with detailed solutions, and percentage with visual elements in columns four to six respectively.

| Subject | Title | Acronym | # Problems | % Solutions | % Visual |
|---|---|---|---|---|---|
| Physics | *Fundamentals of Physics* (Halliday et al., 2013) | `fund` | 142 | 9.2% | 43.0% |
| | *Statistical Thermodynamics* (Engel & Reid, 2010) | `thermo` | 83 | 20.5% | 0.0% |
| | *Classical Dynamics of Particles and Systems* (Thornton & Marion, 2021) | `class` | 66 | 12.1% | 4.5% |
| Chemistry | *Quantum Chemistry* (Levine et al., 2009) | `quan` | 41 | 19.5% | 0.0% |
| | *Quantum Chemistry* (McQuarrie, 2008) | `chemmc` | 47 | 19.1% | 0.0% |
| | *Physical Chemistry* (Atkins et al., 2014a) | `atkins` | 122 | 13.9% | 0.8% |
| | *Physical Chemistry, Quanta, Matter, and Change* (Atkins et al., 2014b) | `matter` | 59 | 16.9% | 3.4% |
| Math | *Calculus: Early Transcendentals* (Stewart et al., 2012) | `calc` | 161 | 19.3% | 67.7% |
| | *Probability and Statistical Inference* (Hogg et al., 1977) | `stat` | 93 | 21.5% | 1.1% |
| | *Elementary Differential Equations and Boundary Value Problems* (Boyce et al., 2021) | `diff` | 55 | 9.1% | 0.0% |

While the aforementioned datasets focus on evaluating LLMs' performance on scientific problem solving tasks, another line of research aims to analyze the diverse capabilities of LLMs more comprehensively. Liu et al. (2023c) assess the reading abilities of LLMs using multiple-choice questions. Frieder et al. (2023) focus on evaluating the mathematical capabilities of LLMs, including those at the college level, but with topics such as functional analysis or topology that differ from those in SciBench, such as differential equations and calculus. Bubeck et al. (2023) explore the comprehensive abilities of GPT-4, but only use up to high-school level mathematical problems such as those in GSM8k (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021). Zhang et al. (2024) develop SciGLM, a scientific language model for collegiate-level problem reasoning, and evaluate its performance across multiple scientific datasets. Kabir et al. (2023) conduct a detailed manual analysis for LLMs. They also provide human-annotated qualitative analysis to assess the capabilities of the models. However, relying on human labor for direct solution analysis can be costly. Our evaluation protocol, based on predefined fundamental problem solving skills, enables automated classification of deficient skills for each incorrectly answered question. This approach enables an affordable, large-scale qualitative analysis of model solutions.

## 3. The SciBench Dataset

To evaluate the capabilities and analyze the limitations of Large Language Models (LLMs) to solve scientific computing problems, we collect a new dataset consisting of college-level textbooks and course exams in a variety of domains. This section details the dataset construction process.

**Data selection criteria.** Our dataset aims to improve the previous benchmarks by including more challenging problems. Specifically, the selected dataset should fulfill the following requirements:

- **Inclusion of college-level problems.** The chosen problems demand a solid understanding of domain-specific knowledge, adept calculation skills, and the ability to perform complex numerical computations.
- **Inclusion of detailed solutions.** To facilitate a thorough analysis of the limitations of LLMs, detailed solutions should be provided as well, which could facilitate a finer-grained examination of the capacity of LLMs to handle complex problem-solving tasks.
- **Inclusion of visual elements.** In the real world, many scientific problems require the interpretation and integration of both textual and visual information. The included problems should thus contain visual elements (such as figures) in the contexts.
- **Inaccessibility in text formats.** To ensure an unbiased evaluation, questions should not be readily accessible online and cannot be easily extracted or transformed into text. This aims to mitigate any potential information leakage from the exposure of LLMs to pre-existing online question banks, such as those found in standardized tests like the SAT exams.
- **Assessment of advanced problem-solving capabilities.** The problems to benchmark should not be confined to basic arithmetic operations like addition and multiplication. Rather, they should enable evaluating the capability of LLMs in performing advanced computations such as calculus and differential equations.

Accordingly, to construct the dataset, we select ten textbooks from three scientific fields Physics, Chemistry, and Mathematics that have been extensively used in college courses. We summarize the statistics of this textbook dataset in Table 2 and we use acronyms to refer to each textbook throughout the paper for brevity. Furthermore, in order to simulate real-world evaluation, we compile a closed set of exam questions from college courses from Computer Science and Math departments, including *Data Mining*, *Machine Learning*, and *Differential Equations*. This subset is less likely to be in LLM training data, making it an effective tool for LLM evaluation. Detailed statistics of these exam problems are summarized in Table S1. We refer readers to Appendix A for details on these textbooks and exams.

To reduce the likelihood of correct answers being merely guessed from candidates, we choose to mainly include questions with more challenging, free-response answers, rather than multiple-choice questions in previous works (Chen et al., 2023b; Lu et al., 2021a; 2022). In order to facilitate standardized and automated evaluation, we focus on answers that only contain single numerical numbers to avoid ambiguity for the textbook dataset. Further, we convert the answer to floating-point numbers rounded to three decimal places. For example, the answer $\frac{\sqrt{2}}{\pi}$ will be converted to the decimal representation of $0.450$. We also treat scientific notation as a unit to avoid overflow issues. For example, if the answer is $2.2 \times 10^{-31}$ m, we take 2.2 as the final answer and $10^{-31}$ m as the unit.

**Data preprocessing.** We collect each problem from the original textbooks in PDF documents and manually process them into LaTeX documents using an OCR tool Mathpix. The data is manually collected by human annotators using a web-based annotation tool (Lu et al., 2021a), whose user interface is shown in Appendix A.3. All problems are carefully verified by human annotators to ensure that LaTeX documents can be compiled without any syntax errors. For reference, we also provide the original numbers in textbooks. For every problem, we provide the answer in two forms: the numerical value and the corresponding LaTeX expression with mathematical notations retained (e.g., $0.450$ and $\frac{\sqrt{2}}{\pi}$); the unit of each answer is saved as a separate attribute. The detailed step-by-step solutions are also provided in LaTeX. For problems having multiple answers, we either keep only the first subproblem and discard the remaining subproblems or convert each subproblem into a separate problem.

## 4. Experiments

This section presents the experiments to assess the capabilities of LLMs in scientific problem-solving. We first describe our experimental setup. Subsequently, we evaluate unimodal LLMs on the textbook dataset. Following this, we include additional experiments on the multimodal subset and the closed exam subset, as well as comparisons with other numerical computational tools.

### 4.1. Experiment Setup

We evaluate the textbook dataset on seven unimodal LLMs, which include four proprietary models: Claude2 (`claude2`) (Anthropic., 2023), GPT-3.5-Turbo (`gpt-3.5-turbo`) (OpenAI., 2022), GPT-4 (`gpt-4`), GPT-4-Turbo (`gpt-4-turbo`) (OpenAI., 2023), along with three open-source models: LLaMA-2-7B (`llama-2-7b-chat`), LLaMA-2-70B (`llama-2-70b-chat`) (Touvron et al., 2023b), and Mistral-7B (`mistral-7b-instruct`) (Jiang et al., 2023).

We consider two prompting strategies, including the Chain-of-Thought (CoT) prompting and prompting to use external tools.

- **Zero-shot and few-shot learning.** In the zero-shot learning setting, models are not provided with any prior examples, which evaluates their inherent problem-solving capabilities with background knowledge and reasoning abilities. In the few-shot setting, a few examples are given to the models before the test example. This aims to assess their capability to learn new information from the demonstrations and incorporate it into their problem-solving processes.
- **Prompting-based approaches.** For our experiments, all settings begin with a system prompt that describes the types and categories of questions. Additionally, we utilize a CoT prompting strategy in zero- and few-shot settings.
- **Tool-augmented approaches.** Given that LLMs are limited in acquiring exact knowledge and performing precise calculations, some recent approaches, such as PAL (Gao et al., 2022) and PoT (Chen et al., 2023a) explore utilizing external tools such as the Python interpreter for program synthesis to enhance the capabilities of solving complex reasoning tasks. In line with these approaches and acknowledging the limitations of LLMs in performing precise calculations, we also include a setting that prompts the model to convert its solution steps in natural language into Python code, aiming to achieve more accurate results for certain computation steps. This tool-augmented approach can only be tested in the few-shot learning setting. We manually construct Python programs that produce the correct answer.

**Implementation details.** We set temperature to zero for all models to reduce the randomness of the predictions. Few-shot examples, including solutions, are randomly selected from problems within each textbook. When external tools are used, we add a code snippet that translates the solution into specific programming languages in all few-shot examples. The code snippets are verified by human annotators that will produce the correct output. In terms of evaluation metrics, we compare the model outputs with the correct answers, allowing a relative tolerance of 5%. In particular to the exam dataset, the model solutions are graded using the rubrics provided by the instructors. Readers may refer to Appendix C for all prompts and the implementation details for utilizing external tools.

### 4.2. Results and Analysis

We report the model performance in terms of accuracy score for each textbook and an average score over all problems. The results of all LLMs in various settings on the textbook and the exam dataset are summarized in Tables 3 and S2 respectively. We have the following observations.

*Table 3.* Experimental results in terms of accuracy (%) on the textbook dataset. The best performing score is highlighted in **bold** and second-best is underlined. The average score is weighted by the number of problems in each textbook.

| Model | Chemistry | | | | Physics | | | Math | | | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | atkins | chemmc | quan | matter | fund | class | thermo | diff | stat | calc | |
| *Zero-Shot Learning* | | | | | | | | | | | |
| LLaMA-2-7B | 0.00 | 0.00 | 0.00 | 0.00 | 1.37 | 0.00 | 0.00 | 2.00 | 5.33 | 0.00 | 1.03 |
| LLaMA-2-70B | 1.87 | 2.56 | 0.00 | 0.00 | 1.40 | 0.00 | 0.00 | 0.00 | 10.70 | 4.76 | 2.41 |
| Mistral-7B | 9.35 | 5.13 | 8.82 | 4.08 | 5.48 | 2.13 | 0.00 | 4.00 | 12.00 | 2.38 | 6.23 |
| Claude2 | 15.00 | 12.83 | 14.71 | 10.20 | 12.33 | 6.40 | 9.00 | 4.00 | 38.70 | 16.70 | 14.94 |
| GPT-3.5-Turbo | 4.67 | 20.51 | 8.82 | 2.04 | 10.96 | 2.13 | 2.94 | 6.00 | 28.00 | 9.30 | 9.59 |
| GPT-4 | <u>45.79</u> | <u>28.21</u> | <u>26.47</u> | <u>22.45</u> | <u>23.29</u> | **25.53** | 17.91 | 32.00 | <u>49.33</u> | **54.76** | <u>33.79</u> |
| GPT-4-Turbo | **57.01** | **41.03** | **35.29** | **26.53** | **24.66** | <u>21.28</u> | **26.87** | **46.00** | **61.33** | <u>52.38</u> | **40.99** |
| *Zero-Shot Learning + CoT Prompting* | | | | | | | | | | | |
| LLaMA-2-7B | 0.00 | 2.56 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 4.00 | 0.00 | 0.67 |
| LLaMA-2-70B | 0.93 | 2.56 | 0.00 | 0.00 | 0.00 | 0.00 | 1.49 | 0.00 | 10.70 | 0.00 | 1.89 |
| Mistral-7B | 6.54 | 5.13 | 2.94 | 0.00 | 0.00 | 2.12 | 1.49 | 6.00 | 10.67 | 9.52 | 4.63 |
| Claude2 | 20.56 | 15.38 | 8.82 | 4.08 | 8.23 | 4.26 | 5.97 | 6.00 | 36.00 | 14.29 | 13.89 |
| GPT-3.5-Turbo | 6.54 | 23.08 | 2.94 | 10.20 | 12.33 | 2.12 | 5.97 | 12.00 | 33.33 | 9.30 | 12.17 |
| GPT-4 | <u>28.04</u> | **43.59** | <u>14.71</u> | <u>20.41</u> | <u>21.92</u> | <u>19.15</u> | 17.91 | 22.00 | <u>50.67</u> | <u>42.86</u> | 28.52 |
| GPT-4-Turbo | **60.75** | <u>35.90</u> | **29.41** | **28.57** | **30.14** | **31.91** | 25.37 | 38.00 | **64.00** | **54.76** | **42.37** |
| *Few-Shot Learning + CoT Prompting* | | | | | | | | | | | |
| LLaMA-2-7B | 1.87 | 5.13 | 2.94 | 0.00 | 5.48 | 0.00 | 0.00 | 0.00 | 12.00 | 7.14 | 3.60 |
| LLaMA-2-70B | 13.10 | 12.83 | 14.71 | 4.08 | 12.33 | 0.00 | 0.00 | 0.00 | 13.30 | 9.52 | 8.40 |
| Mistral-7B | 6.54 | 10.26 | 2.94 | 2.04 | 2.74 | 2.13 | 4.48 | 4.00 | 14.67 | 9.52 | 6.17 |
| Claude2 | 15.89 | 25.64 | 14.65 | 6.12 | 9.59 | 6.38 | 10.45 | 8.00 | 33.33 | 19.05 | 15.26 |
| GPT-3.5-Turbo | 8.41 | 20.51 | 8.82 | 6.12 | 10.96 | 2.12 | 1.49 | 10.00 | 38.67 | 6.98 | 11.99 |
| GPT-4 | <u>41.12</u> | <u>33.33</u> | <u>17.65</u> | <u>16.33</u> | <u>17.81</u> | <u>17.02</u> | 20.90 | 30.00 | <u>49.33</u> | <u>45.24</u> | <u>30.36</u> |
| GPT-4-Turbo | **59.81** | **35.90** | **26.47** | **18.37** | **23.29** | **19.15** | **32.84** | **32.00** | **65.33** | **50.00** | **39.45** |
| *Few-Shot Learning + Python* | | | | | | | | | | | |
| LLaMA-2-7B | 0.93 | 2.56 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 6.67 | 0.00 | 1.20 |
| LLaMA-2-70B | 0.93 | 7.69 | 2.94 | 0.00 | 9.59 | 0.00 | 1.49 | 0.00 | 17.30 | 9.52 | 5.14 |
| Mistral-7B | 4.67 | 0.00 | 5.88 | 2.04 | 2.74 | 2.13 | 0.00 | 4.00 | 17.33 | 11.90 | 5.32 |
| Claude2 | 6.54 | 12.82 | 14.71 | 4.08 | 17.81 | 8.51 | 5.97 | 20.00 | 40.00 | 16.67 | 14.92 |
| GPT-3.5-Turbo | 13.08 | <u>33.33</u> | 8.82 | 16.33 | 26.01 | 4.26 | 7.46 | 16.00 | <u>44.00</u> | 26.19 | 19.91 |
| GPT-4 | **57.01** | **38.46** | **44.12** | **34.69** | **28.77** | **23.40** | **34.33** | **44.00** | **68.00** | **38.10** | **43.22** |
| GPT-4-Turbo | <u>32.71</u> | <u>33.33</u> | <u>17.65</u> | <u>26.53</u> | <u>27.40</u> | <u>12.76</u> | 16.42 | 34.00 | 42.67 | <u>30.95</u> | <u>28.47</u> |

- **Observation 1. SCIBENCH is complex enough to differentiate among LLMs.** Our results show that open-source models such as LLaMA-2 and Mistral are consistently outperformed by their proprietary counterparts across all settings within the textbook dataset. Notably, GPT-4 and GPT-4-Turbo lead in performance by a significant margin. For example, GPT-4-Turbo outperforms Mistral-7B by 34.76% in the zero-shot setting. Additionally, within both LLaMA and GPT series, we observe a clear correlation between increased model capacity (i.e., larger parameter sizes) and improved performance. Therefore, the complexity of SCIBENCH is able to differentiate the performance among different LLMs.
- **Observation 2. SCIBENCH highlights varied efficacy of prompting strategies across LLMs.** Our findings suggest that the effectiveness of employing prompting strategies or external computational tools varies signif-

icantly among different LLMs. As shown in the table, LLaMA-2-70B shows a marked improvement in the few-shot setting over the zero-shot setting, increasing from 2.41% to 8.40%. Similarly, the performance of GPT-4 is significantly improved when incorporating external tools, with an increase from 30.36% to 43.22%. Meanwhile, the up-to-date model GPT-4-Turbo exhibits superior performance in zero-shot learning settings. However, despite its advanced capabilities demonstrated by its outstanding zero-shot learning performance, it falls short compared to GPT-4 in few-shot learning when leveraging Python for numerical computation. This suggests a potential reduction in its program understanding capabilities. In summary, such findings illustrate SCIBENCH can reveal the nuanced differences in the ability of LLMs to utilize prompting strategies and external tools effectively.
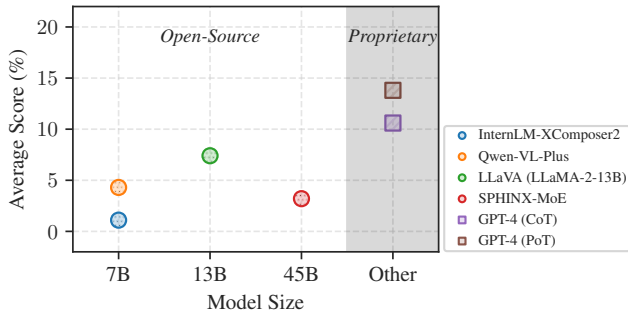
*Figure 2.* Performance of LLMs on the multimodal subset. GPT-4 models are augmented with image captions and OCR text.

### 4.3. Additional Experiments

**Evaluation on the multimodal subset.** We evaluate two categories of models on problems with visual contexts: (1) GPT-4 (OpenAI., 2023) augmented with image captions from Multimodal Bard (Google, 2023) and OCR texts from EasyOCR (JaidedAI, 2022) and (2) open-source Large Multimodal Models (LMMs): InternLM-XComposer2-VL (Dong et al., 2024), Qwen-VL-Plus (Bai et al., 2023), SPHINX-MoE (Lin et al., 2023), and LLaVA-LLaMA-2-13B (Liu et al., 2023a). For GPT-4, we explore two prompting strategies: Chain-of-Thought (CoT) (Wei et al., 2022) and Program-of-Thoughts (PoT) (Chen et al., 2023a). The results presented in Figure 2 reveal that proprietary models augmented with image captions and OCR-detected text, significantly outperform their open-source counterparts. GPT-4 (PoT) that combines programming capabilities achieves an accuracy of 13.8%, markedly higher than 7.4% obtained by the best open model LLaVA-LLaMA-2-13B. This demonstrates the substantial potential for LLMs to effectively utilize visual contexts in scientific problem solving.

**Evaluation on the exam subset.** To mirror real-world testing conditions with no few-shot examples provided, we evaluate GPT-3.5, GPT-4, Claude, LLaMA-2-7B, and LLaMA-2-70B on the closed exam dataset under zero-shot and zero-shot CoT settings. The experiment results summarized in Table S2 indicate a notable performance advantage of GPT-4, which achieves an averaged score of 57.54%. However, we note that their performance remains significantly lower than human benchmarking. For instance, in the Data Mining course, GPT-4 scores 64.44% and 42.67% in the midterm and final exams, lower than the average student scores of 80.18% and 72.71%, respectively, as reported by the course instructor. The results once again underline the challenging nature of our dataset.

**Comparison with other scientific computing tools.** We further utilize another famous scientific computing library Wolfram Language as the external tool and conduct experiments using GPT-3.5, Claude, LLaMA-2-7B, and LLaMA-2-70B. The experiment results reported in Figure S7 show

that utilizing Wolfram Language does not help few-shot learning and even results in a deteriorated performance, with a decrease of 6.70% compared to the CoT prompting for Claude2, and a decrease of 6.17% for LLaMA-2-70B. A plausible explanation is the introduction of syntax errors when translating solution steps into the Wolfram Language, which could be a potential direction for improvement. For a detailed error analysis, readers are directed to Appendix C.3.

## 5. Error Analysis of Prompting Strategies

Considering the substantial advancements of current LLMs, an in-depth analysis of the particular skills that are either enhanced or limited under certain settings becomes imperative. Previous works have relied on human labor to annotate error reasons into different categories, which is both expensive and time-consuming (Zhong et al., 2023). In this section, we present an evaluation protocol that automates the classification of error reasons into deficient skills. This time-efficient approach enables large-scale analyses in future research.

In order to quantify the impact of each setting on scientific problem-solving, we first define an essential skill set that is required by solving scientific problems. Then, an LLM verifier is employed to automatically classify each incorrectly solved problem based on the absence of a specific skill from the essential skill set. This approach generates error profiles, showcasing a direct comparison of different strategies. This evaluation protocol is summarized in Figure 3.

Firstly, we analyze the incorrect solutions made by GPT-3.5 for problems that provide detailed solutions. We hire two college students, who are highly familiar with the problems in our datasets, to annotate the source of the error for each problem, indicating the specific line where the model makes a mistake and why. From 112 such error annotations and with the assistance of GPT-4, we distill these errors into ten essential skills that GPT-3.5 might lack:

- **Logical decomposition and analysis skills.** This ability involves decomposing the problem into smaller, manageable parts, and understanding the relationships between these parts.
- **Assumption identification.** This skill involves the ability to recognize relevant and necessary assumptions in the problem.
- **Spatial perception.** This is important for understanding problems in areas such as Physics and Chemistry, where models need to visualize molecules, forces, fields, etc.
- **Causal reasoning.** This is the ability to understand cause and effect relationships.
- **Problem deduction skills.** This pertains to the ability to infer and deduce potential solutions or underlying principles from the given information in a problem.
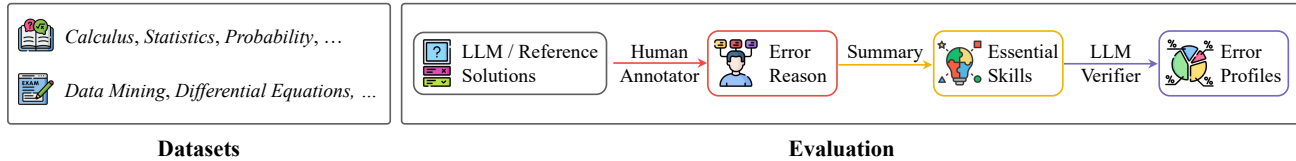
**Datasets** **Evaluation**

*Figure 3.* Pipeline of the evaluation protocol. The evaluation protocol involves analyzing both LLMs and reference (correct) solutions with the assistance of human annotators to identify error reasons. These reasons are then summarized into ten essential scientific problem-solving skills in which LLM may face challenges. Subsequently, a LLM verifier is employed to automatically attribute each incorrectly answered problem to a lack of a specific skill. The resulting error profiles enable the interpretation of the improved skills by certain prompting strategies and the direct comparison of various strategies.
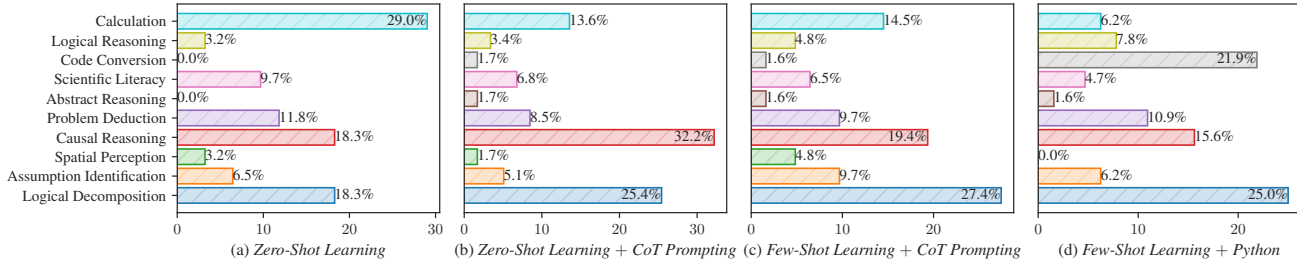


*Figure 4.* Error profiles of GPT-3.5 on the textbook dataset under four settings, which reveal the distribution of their deficiencies in ten essential problem-solving abilities.

- **Abstract reasoning.** This skill involves the ability to understand complex concepts that cannot be perceived physically, and to recognize patterns or relationships beyond concrete examples.
- **Scientific literacy.** This skill involves a comprehensive understanding of key scientific principles, terminology, and methodologies across a range of disciplines.
- **Code conversion skills.** This involves the ability to accurately translate solution steps into different programming languages, like Python or Wolfram Language.
- **Logical reasoning.** This is the ability to make a reasoned argument and to identify fallacies or inconsistencies in an argument or set of data.
- **Calculation skills.** This involves the ability to accurately carry out mathematical operations and computations.

After identifying this essential skill set, we assess the performance of the LLMs under different settings to discern the specific problem-solving skills they lack. Given the high cost of human annotations required to attribute the cause of incorrect solutions to specific skill deficiencies, we propose a novel self-critique protocol: we design a specific prompt that outlines these abilities, and employ another LLM to serve as a classifier and determine whether a specific error results from the lack of a particular problem-solving skill. Finally, we ask human annotators to scrutinize the classification results, which results in approximately 20% of incorrectly classified skills being discarded. To be specific, we utilize a GPT-3.5 model as the verifier to determine the reason behind each error and pinpoint the missing skill. The details regarding the specific prompts used are provided in

Appendix C.1. This verification process is conducted for four settings, with results represented in bar charts (Figure 4). Additional examples of the evaluation protocol are elaborated in Appendix D.

Our findings suggest that **there is a lack of a universally effective setting: each configuration only enhances some specific abilities and occasionally even hurts other skills that the original model possesses.** First, CoT prompting significantly improves calculation skills in the zero-shot scenario, with 13.6% error rates caused by calculation ability, considerably lower than the 29.0% error rate of the vanilla zero-shot baseline. However, CoT shows limitations in improving other skills, with 32.2% and 25.4% error rates in casual ability and logical decomposition ability in the zero-shot CoT setting, respectively, compared to 18.3% and 18.3% in the zero-shot setting. This contradicts previous claims about universal skill enhancement through zero-shot CoT and carefully-designed few-shot CoT prompts (Wei et al., 2022). An example in Figure S9 shows that the zero-shot learning setting without CoT has generated the correct formula but fails in the calculation steps. In this case, CoT prompting is even unable to use the correct formula as it misinterprets the specific conditions (non-necessity) in the problem. Second, the use of external tools significantly reduces calculation errors compared to the few-shot Cot setting, with a notable decrease from 14.5% to 6.2%. However, the use of external tools can weaken other skills, particularly the code conversion skills, i.e., generating the correct programs for the solution. Third, few-shot learning does not universally improve scientific problem-solving skills, as

indicated in the comparison between zero-shot and few-shot CoT settings. The improvement in one skill is offset by the shortcomings in others: although the few-shot CoT setting results in a reduction of 12.8% in errors related to causal reasoning, it also leads to an increase in errors associated with other skills, such as logical decomposition.

## 6. Conclusion

This paper presents SciBench, a college-level benchmark that includes scientific problems from Mathematics, Physics, and Chemistry, as well as exam questions in Computer Science and Mathematics. Our comprehensive evaluation includes a diverse array of Large Language Models (LLMs), spanning both open-source and proprietary models, including unimodal as well as multimodal settings, and employing a variety of prompting strategies. The evaluation protocol we employ serves as a framework for evaluating advanced problem-solving skills of LLMs in scientific domains. The findings of this study highlight that while large language models (LLMs) exhibit impressive performance on introductory mathematical benchmarks, their mastery of problem solving ability remains weak. These findings underscore the limitations of current LLMs in achieving satisfactory performance, even with the assistance of various tools. We envision that the SciBench benchmark dataset and evaluation protocol presented in this paper could lay a foundation for future research and enable advancements in understanding and enhancing problem-solving capabilities of LLMs.

## Reproducibility Statement

To foster reproducible research, we include all dataset processing and experiment details of SciBench. We detail data processing in Section 3 and provide the UI design of data collection in Appendix A.3. We include all experiment details with LLM prompts in Appendix C. Finally, we make our dataset and code publicly available at this repository.

## Ethical Statement

The questions of SciBench are sourced from science textbooks and exams. We conduct a manual examination of our dataset to ensure the absence of potential sensitive background or ethical concerns. The inclusion of exam questions has been authorized by the instructors of the respective courses.

The purpose of the textbook dataset is solely for academic use. Its collection adheres to the *Fair Use Law* in the US, where only a certain number of questions from each textbook are selected, ensuring that only a small portion of the textbook is utilized.

## Impact Statement

The introduction of SciBench represents a significant advancement in the evaluation of Large Language Models (LLMs) for scientific problem-solving tasks. By focusing on collegiate-level problems in mathematics, chemistry, and physics, SciBench addresses a critical gap in existing benchmarks, which have primarily focused on high-school subjects and basic algebraic operations. This development underscores the necessity of developing specialized benchmarks that challenge LLMs with higher complexity problems, thereby pushing the boundaries of the capabilities of LLMs in academic and research settings.

While the current scope of SciBench encompasses a select group of scientific disciplines, the potential for future extensions is vast. Incorporating additional subjects such as biology, computer science, and engineering could provide a more comprehensive understanding of LLM capabilities across a broader spectrum of scientific knowledge. Moreover, extending the benchmark to social sciences, humanities, and other human-centric domains would be equally beneficial, as these areas often involve nuanced reasoning and interpretation of complex social dynamics and ethical considerations, posing unique challenges that could further enhance the versatility and applicability of LLMs.

## Acknowledgements

## References

Anthropic. Claude2. https://www.anthropic.com/index/claude-2, 2023. 5

Arora, D., Singh, H. G., et al. Have llms advanced enough? a challenging problem solving benchmark for large language models. *arXiv preprint arXiv:2305.15074*, 2023. 1, 3

Atkins, P., Atkins, P. W., and de Paula, J. *Atkins' physical chemistry*. Oxford university press, 2014a. 4, 12

Atkins, P., De Paula, J., and Friedman, R. *Physical chemistry: quanta, matter, and change*. Oxford University Press, USA, 2014b. 2, 4, 12

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. 7

Boyce, W. E., DiPrima, R. C., and Meade, D. B. *Elementary differential equations and boundary value problems*. John Wiley & Sons, 2021. 4, 12

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 1

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023. 4

Chen, M., Tworek, J., Jun, H., Yuan, Q., Pinto, H. P. d. O., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021. 1

Chen, W., Ma, X., Wang, X., and Cohen, W. W. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research (TMLR)*, 2023a. 1, 5, 7

Chen, W., Yin, M., Ku, M., Lu, P., Wan, E., Ma, X., Xu, J., Xia, T., and Wang, X. Theoremqa: A theorem-driven question answering dataset. *arXiv preprint arXiv:2305.12524*, 2023b. 3, 5

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021. 1, 3, 4

Dong, X., Zhang, P., Zang, Y., Cao, Y., Wang, B., Ouyang, L., Wei, X., Zhang, S., Duan, H., Cao, M., Zhang, W., Li, Y., Yan, H., Gao, Y., Zhang, X., Li, W., Li, J., Chen, K., He, C., Zhang, X., Qiao, Y., Lin, D., and Wang, J. Internlm-xcomposer2: Mastering free-form text-image composition and comprehension in vision-language large model. *arXiv preprint arXiv:2401.16420*, 2024. 7

Engel, T. and Reid, P. J. *Thermodynamics, statistical thermodynamics, and kinetics*. Prentice Hall Upper saddle River, 2010. 4, 12

Frieder, S., Pinchetti, L., Griffiths, R.-R., Salvatori, T., Lukasiewicz, T., Petersen, P. C., Chevalier, A., and Berner, J. Mathematical capabilities of chatgpt. *arXiv preprint arXiv:2301.13867*, 2023. 4

Fu, Y., Ou, L., Chen, M., Wan, Y., Peng, H., and Khot, T. Chain-of-thought hub: A continuous effort to measure large language models' reasoning performance. *arXiv preprint arXiv:2305.17306*, 2023. 3

Gao, L., Madaan, A., Zhou, S., Alon, U., Liu, P., Yang, Y., Callan, J., and Neubig, G. PAL: Program-aided language models. *arXiv preprint arXiv:2211.10435*, 2022. 1, 5

Gao, P., Han, J., Zhang, R., Lin, Z., Geng, S., Zhou, A., Zhang, W., Lu, P., He, C., Yue, X., Li, H., and Qiao, Y. Llama-adapter v2: Parameter-efficient visual instruction model. *arXiv preprint arXiv:2304.15010*, 2023. 1

Ghazal, A., Rabl, T., Hu, M., Raab, F., Poess, M., Crolotte, A., and Jacobsen, H.-A. Bigbench: Towards an industry standard benchmark for big data analytics. In *Proceedings of the 2013 ACM SIGMOD international conference on Management of data*, pp. 1197–1208, 2013. 3

Google. Bard. `https://bard.google.com`, 2023. 7

Guo, T., Guo, K., Liang, Z., Guo, Z., Chawla, N. V., Wiest, O., Zhang, X., et al. What indeed can gpt models do in chemistry? a comprehensive benchmark on eight tasks. *arXiv preprint arXiv:2305.18365*, 2023. 3

Halliday, D., Resnick, R., and Walker, J. *Fundamentals of physics*. John Wiley & Sons, 2013. 4, 12

Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., and Steinhardt, J. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020. 1, 3

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021. 1, 3, 4

Hogg, R. V., Tanis, E. A., and Zimmerman, D. L. *Probability and statistical inference*, volume 993. Macmillan New York, 1977. 4, 13

Huang, J., Gu, S. S., Hou, L., Wu, Y., Wang, X., Yu, H., and Han, J. Large language models can self-improve. *arXiv preprint arXiv:2210.11610*, 2022. 1

JaidedAI. Easyocr: Ready-to-use ocr. `https://github.com/JaidedAI/EasyOCR`, 2022. 7

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 5

Kabir, S., Udo-Imeh, D. N., Kou, B., and Zhang, T. Who answers it better? an in-depth analysis of chatgpt and stack overflow answers to software engineering questions. *arXiv preprint arXiv:2308.02312*, 2023. 4

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*, 2022. 1

Levine, I. N., Busch, D. H., and Shull, H. *Quantum chemistry*, volume 6. Pearson Prentice Hall Upper Saddle River, NJ, 2009. 4, 12

Lin, Z., Liu, C., Zhang, R., Gao, P., Qiu, L., Xiao, H., Qiu, H., Lin, C., Shao, W., Chen, K., et al. Sphinx: The joint mixing of weights, tasks, and visual embeddings for multi-modal large language models. *arXiv preprint arXiv:2311.07575*, 2023. 7

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *NeurIPS*, 2023a. 7

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*, 2023b. 1

Liu, H., Ning, R., Teng, Z., Liu, J., Zhou, Q., and Zhang, Y. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*, 2023c. 4

Lu, P., Gong, R., Jiang, S., Qiu, L., Huang, S., Liang, X., and Zhu, S.-C. Inter-gps: Interpretable geometry problem solving with formal language and symbolic reasoning. In *The Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*, 2021a. 5

Lu, P., Qiu, L., Chen, J., Xia, T., Zhao, Y., Zhang, W., Yu, Z., Liang, X., and Zhu, S.-C. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. *arXiv preprint arXiv:2110.13214*, 2021b. 3

Lu, P., Mishra, S., Xia, T., Qiu, L., Chang, K.-W., Zhu, S.-C., Tafjord, O., Clark, P., and Kalyan, A. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022. 1, 3, 5

Lu, P., Bansal, H., Xia, T., Liu, J., Li, C., Hajishirzi, H., Cheng, H., Chang, K.-W., Galley, M., and Gao, J. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023a. 3

Lu, P., Peng, B., Cheng, H., Galley, M., Chang, K.-W., Wu, Y. N., Zhu, S.-C., and Gao, J. Chameleon: Plug-and-play compositional reasoning with large language models. *arXiv preprint arXiv:2304.09842*, 2023b. 1

Lu, P., Qiu, L., Chang, K.-W., Wu, Y. N., Zhu, S.-C., Rajpurohit, T., Clark, P., and Kalyan, A. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations (ICLR)*, 2023c. 3

Lu, P., Qiu, L., Yu, W., Welleck, S., and Chang, K.-W. A survey of deep learning for mathematical reasoning. In *The 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023d. 3

McQuarrie, D. A. *Quantum chemistry*. University Science Books, 2008. 4, 12

Mishra, S., Finlayson, M., Lu, P., Tang, L., Welleck, S., Baral, C., Rajpurohit, T., Tafjord, O., Sabharwal, A., Clark, P., et al. Lila: A unified benchmark for mathematical reasoning. In *The 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022. 3

OpenAI. Chatgpt: Optimizing language models for dialogue. https://openai.com/blog/chatgpt/., 2022. 1, 5

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 5, 7

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*, 2023. 1

Stewart, J., Watson, S., and Clegg, D. Calculus: Early transcendentals, 8th. *Edition, Brooks/Cole, Cengae learning*, 2012. 4, 13

Sun, L., Han, Y., Zhao, Z., Ma, D., Shen, Z., Chen, B., Chen, L., and Yu, K. Scieval: A multi-level large language model evaluation benchmark for scientific research. *arXiv preprint arXiv:2308.13149*, 2023. 3

Suzgun, M., Scales, N., Schärli, N., Gehrmann, S., Tay, Y., Chung, H. W., Chowdhery, A., Le, Q. V., Chi, E. H., Zhou, D., et al. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*, 2022. 3

Taylor, R., Kardas, M., Cucurull, G., Scialom, T., Hartshorn, A., Saravia, E., Poulton, A., Kerkez, V., and Stojnic, R. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*, 2022. 3

Thornton, S. T. and Marion, J. B. *Classical dynamics of particles and systems*. Cengage Learning, 2021. 4, 12

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a. 1

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b. 5

Wang, X., Wei, J., Schuurmans, D., Le, Q., Chi, E., and Zhou, D. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022. 1

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Chi, E., Le, Q., and Zhou, D. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022. 1, 7, 8

Welleck, S., Liu, J., Bras, R. L., Hajishirzi, H., Choi, Y., and Cho, K. Naturalproofs: Mathematical theorem proving in natural language. *arXiv preprint arXiv:2104.01112*, 2021. 3

Zhang, D., Hu, Z., Zhoubian, S., Du, Z., Yang, K., Wang, Z., Yue, Y., Dong, Y., and Tang, J. Sciglm: Training scientific language models with self-reflective instruction annotation and tuning. *arXiv preprint arXiv:2401.07950*, 2024. 4

Zhang, R., Han, J., Zhou, A., Hu, X., Yan, S., Lu, P., Li, H., Gao, P., and Qiao, Y. Llama-adapter: Efficient fine-tuning of language models with zero-init attention. *arXiv preprint arXiv:2303.16199*, 2023a. 1

Zhang, Z., Zhang, A., Li, M., Zhao, H., Karypis, G., and Smola, A. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*, 2023b. 1

Zhong, W., Cui, R., Guo, Y., Liang, Y., Lu, S., Wang, Y., Saied, A., Chen, W., and Duan, N. Agieval: A human-centric benchmark for evaluating foundation models. *arXiv preprint arXiv:2304.06364*, 2023. 1, 3, 7

Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Bousquet, O., Le, Q., and Chi, E. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*, 2022. 1

# Supplementary Material for SCIBENCH

## A. The Textbook Dataset

### A.1. Textbook Sources

- PHYSICAL CHEMISTRY (ATKINS ET AL., 2014A) (`atkins`) provides an exploration of equilibrium, structure, and reactions, integrating contemporary techniques like nanoscience, spectroscopy, and computational chemistry.
- QUANTUM CHEMISTRY (MCQUARRIE, 2008) (`chemmc`) meticulously covers Quantum Mechanics, from foundational principles like blackbody radiation and Heisenberg's Uncertainty Principle to complex topics such as Schrödinger equation, quantum mechanical operators, and the application of quantum mechanics in chemical bonding.
- QUANTUM CHEMISTRY (LEVINE ET AL., 2009) (`quan`) explores quantum chemistry, providing a detailed understanding of the Schrödinger equation, particle behavior in various scenarios, quantum mechanics operators, and other foundational quantum principles. It delves into specific applications like the electronic structure of diatomic and polyatomic molecules, variation methods, perturbation theory, electron spin and its implications in quantum mechanics, as well as various computational methods for molecular quantum mechanics.
- PHYSICAL CHEMISTRY, QUANTA, MATTER, AND CHANGE (ATKINS ET AL., 2014B) (`matter`) combines physics and mathematics, beginning with basics like differentiation and integration, advancing through quantum mechanics and atomic structure, then exploring thermodynamics, molecular motion, and chemical kinetics. Each section is supplemented with mathematical concepts such as differential equations, vectors, and probability theory.
- CLASSICAL DYNAMICS OF PARTICAL AND SYSTEMS (THORNTON & MARION, 2021) (`class`) initiates with an exploration of fundamental mathematical concepts, discussing scalars, vectors, matrix operations, coordinate transformations, differentiation, and integration of vectors, using these constructs to illustrate concepts like velocity, acceleration, and angular velocity. It then transitions into the realm of Newtonian mechanics, detailing Newton's laws, frames of reference, and the equation of motion for a single particle.
- THERMODYNAMICS, STATISTICAL THERMODYNAMICS, AND KINETICS (ENGEL & REID, 2010) (`thermo`) navigates through thermodynamics' principles, from fundamental concepts to complex laws, further discussing real and ideal gases, solutions, electrochemical cells, and statistical thermodynamics. It concludes with an examination of the kinetic theory of gases, transport phenomena, and chemical kinetics.
- FUNDAMENTALS OF PHYSICS (HALLIDAY ET AL., 2013) (`fund`) covers undergraduate physics topics, ranging from fundamental concepts like motion and energy to more advanced areas such as quantum physics and nuclear physics.
- ELEMENTARY DIFFERENTIAL EQUATIONS AND BOUNDARY VALUE PROBLEMS (BOYCE ET AL., 2021) (`diff`) provides a detailed exploration of differential equations, progressing from basic mathematical models to advanced topics

like the Laplace Transform, linear systems, numerical methods, and Fourier series. It culminates with a deep dive into nonlinear equations, partial differential equations, and boundary value problems.

- PROBABILITY AND STATISTICAL INFERENCE (HOGG ET AL., 1977) (stat) covers probability and statistics, including fundamental concepts, discrete and continuous distributions, bivariate distributions, functions of random variables, and estimation techniques.
- CALCULUS: EARLY TRANSCENDENTALS (STEWART ET AL., 2012) (calculus) begins with diagnostic tests in foundational topics, and explores functions from multiple perspectives. It comprehensively covers calculus concepts from limits to three-dimensional analytic geometry, incorporating applications in various fields.

## A.2. Textbook Examples

The textbook examples are provided in Figure S1. The examples from the multimodal subset are provided in Figures S2 to S5.

---

**Problem** (fund)
Two charged particles are fixed to an $x$ axis: Particle 1 of charge $q_1 = 2.1 \times 10^{-8}$C is at position $x = 20$ cm and particle 2 of charge $q_2 = -4.00q_1$ is at position $x = 70$ cm. At what coordinate on the axis (other than at infinity) is the net electric field produced by the two particles equal to zero?
**Answer:** $-30$ cm

---

**Problem** (thermo)
$N_2O_3$ dissociates according to the equilibrium $N_2O_3(g) \rightleftharpoons NO_2(g) + NO(g)$. At 298 K and one bar pressure, the degree of dissociation defined as the ratio of moles of $NO_2(g)$ or $NO(g)$ to the moles of the reactant assuming no dissociation occurs is $3.5 \times 10^{-3}$. Calculate $\Delta G_R^\circ$ for this reaction.
**Answer:** 28 kJ mol$^{-1}$

---

**Problem** (class)
Halley's comet, which passed around the sun early in 1986, moves in a highly elliptical orbit with an eccentricity of 0.967 and a period of 76 years. Calculate its minimum distances from the Sun.
**Answer:** $8.8 \times 10^{10}$m

---

**Problem** (quan)
A one-particle, one-dimensional system has $\Psi = a^{-1/2}e^{-|x|/a}$ at $t = 0$, where $a = 1.0000$ nm. At $t = 0$, the particle's position is measured. Find the probability that the measured value is between $x = 0$ and $x = 2$ nm.
**Answer:** 0.4908

---

**Problem** (chemmc)
One of the most powerful modern techniques for studying structure is neutron diffraction. This technique involves generating a collimated beam of neutrons at a particular temperature from a high-energy neutron source and is accomplished at several accelerator facilities around the world. If the speed of a neutron is given by $v_n = (3k_BT/m)^{1/2}$, where $m$ is the mass of a neutron, then what temperature is needed so that the neutrons have a de Broglie wavelength of 50pm ?
**Answer:** 2500 K

---

**Problem** (atkins)
The change in molar internal energy when $CaCO_3(s)$ as calcite converts to another form, aragonite, is $+0.21$ kJ mol$^{-1}$. Calculate the difference between the molar enthalpy and internal energy changes when the pressure is 1.0 bar given that the densities of the polymorphs are 2.71 g cm$^{-3}$ and 2.93 g cm$^{-3}$, respectively.
**Answer:** -0.28 Pa m$^3$ mol$^{-1}$

---

**Problem** (matter)
In an industrial process, nitrogen is heated to 500 K at a constant volume of 1.000 m$^3$. The gas enters the container at 300 K and 100 atm. The mass of the gas is 92.4 kg. Use the van der Waals equation to determine the approximate pressure of the gas at its working temperature of 500 K. For nitrogen, $a = 1.39$dm$^6$ atm mol$^{-2}$, $b = 0.0391$dm$^3$ mol$^{-1}$.
**Answer:** 140 atm

---

**Problem** (calc)
A planning engineer for a new alum plant must present some estimates to his company regarding the capacity of a silo designed to contain bauxite ore until it is processed into alum. The ore resembles pink talcum powder and is poured from a conveyor at the top of the silo. The silo is a cylinder 100ft high with a radius of 200ft. The conveyor carries ore at a rate of $60,000\pi$ ft$^3$/h and the ore maintains a conical shape whose radius is 1.5 times its height. If, at a certain time $t$, the pile is 60ft high, how long will it take for the pile to reach the top of the silo?
**Answer:** 9.8 h

---

**Problem** (stat)
In a study concerning a new treatment of a certain disease, two groups of 25 participants in each were followed for five years. Those in one group took the old treatment and those in the other took the new treatment. The theoretical dropout rate for an individual was 50% in both groups over that 5-year period. Let $X$ be the number that dropped out in the first group and $Y$ the number in the second group. Assuming independence where needed, give the sum that equals the probability that $Y \geq X + 2$. HINT: What is the distribution of $Y - X + 25$ ?
**Answer:** 0.3359

---

**Problem** (diff)
Newton's law of cooling states that the temperature of an object changes at a rate proportional to the difference between its temperature and that of its surroundings. Suppose that the temperature of a cup of coffee obeys Newton's law of cooling. If the coffee has a temperature of $200°$F when freshly poured, and 1 min later has cooled to $190°$F in a room at $70°$F, determine when the coffee reaches a temperature of $150°$F
**Answer:** 6.07 min

---

*Figure S1.* Textbook examples with acronym highlighted in brown.

**Problem**

The region $\mathscr{R}$ enclosed by the curves $y = x$ and $y = x^2$ is rotated about the $x$-axis. Find the volume of the resulting solid.

**Image**



**Correct Solution**

The curves $y = x$ and $y = x^2$ intersect at the points $(0, 0)$ and $(1, 1)$. The region between them, the solid of rotation, and a cross-section perpendicular to the $x$-axis are shown in the Figure. A cross-section in the plane $P_x$ has the shape of a washer (an annular ring) with inner radius $x^2$ and outer radius $x$, so we find the cross-sectional area by subtracting the area of the inner circle from the area of the outer circle:

$$A(x) = \pi x^2 - \pi \left( x^2 \right)^2 = \pi \left( x^2 - x^4 \right)$$

Therefore we have

$$V = \int_0^1 A(x) dx = \int_0^1 \pi \left( x^2 - x^4 \right) dx$$

$$= \pi \left[ \frac{x^3}{3} - \frac{x^5}{5} \right]_0^1 = \frac{2\pi}{15}$$

**Final Answer:** $\frac{2\pi}{15}$

*Figure S2.* The example from the textbook *Calculus: Early Transcendentals*.

### A.3. UI Design of the Labeling Tool

We employed a team of seven individuals to gather data from textbooks using an annotation tool. Each individual was responsible for one to two books, encompassing approximately 100 examples. The user interface of the annotation tool is depicted in Figure S6. For subsequent verification, we preserved images of problems and their corresponding answers. To ensure clarity in future references, we have maintained the original sequence of problems as they appear in the textbooks.

## B. The Exam Dataset

The exam dataset is drawn from the following sources:

- INTRODUCTION TO DATA MINING provides an introductory survey of data mining, which involves the automatic discovery of patterns, associations, changes, and anomalies in large databases. It explores various application areas of data mining, including bioinformatics, e-commerce, environmental studies, financial markets, multimedia data processing, network monitoring, and social service analysis.
- FUNDAMENTALS ARTIFICIAL INTELLIGENCE provides an introduction to the core problem-solving and knowledge representation paradigms in artificial intelligence. It covers Lisp programming with regular assignments, as well as topics such as search methods, planning techniques, knowledge structures, natural language processing, expert systems, vision, and parallel architectures.
- DIFFERENTIAL EQUATIONS covers various topics in differential equations, including first-order and second-order linear equations with constant coefficients, power series solutions, and linear systems. Students will explore the principles and applications of these mathematical concepts.

A detailed statistics of the exam dataset is summarized in Table S1. The experiment results of exam dataset are provided in Table S2.

**Problem**

A 2.00 kg particle moves along an $x$ axis in one-dimensional motion while a conservative force along that axis acts on it. The potential energy $U(x)$ associated with the force is plotted in the Figure. That is, if the particle were placed at any position between $x = 0$ and $x = 7.00$ m, it would have the plotted value of $U$. At $x = 6.5$ m, the particle has velocity $\vec{v}_0 = (-4.00 \text{ m/s})\hat{i}$. From the Figure, determine the particle's speed at $x_1 = 4.5$ m.

**Image**
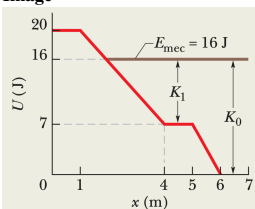


**Correct Solution**

The particle's kinetic energy is given by Eq $\left(K = \frac{1}{2}mv^2\right)$. Because only a conservative force acts on the particle, the mechanical energy $E_{\text{mec}}(= K + U)$ is conserved as the particle moves. Therefore, on a plot of $U(x)$, the kinetic energy is equal to the difference between $E_{\text{mec}}$ and $U$.

Calculations: At $x = 6.5$ m, the particle has kinetic energy

$$K_0 = \frac{1}{2}mv_0^2 = \frac{1}{2}(2.00 \text{ kg})(4.00 \text{ m/s})^2 \tag{S1}$$

$$= 16.0 \text{ J}. \tag{S2}$$

Because the potential energy there is $U = 0$, the mechanical energy is $E_{\text{mec}} = K_0 + U_0 = 16.0 \text{ J} + 0 = 16.0 \text{ J}$. This value for $E_{\text{mec}}$ is plotted as a horizontal line in the Figure. From that figure we see that at $x = 4.5$ m, the potential energy is $U_1 = 7.0$ J. The kinetic energy $K_1$ is the difference between $E_{\text{mec}}$ and $U_1$ :

$$K_1 = E_{\text{mec}} - U_1 = 16.0 \text{ J} - 7.0 \text{ J} = 9.0 \text{ J}. \tag{S3}$$

Because $K_1 = \frac{1}{2}mv_1^2$, we find $v_1 = 3.0$ m/s.
**Final Answer:** 3.0 m/s

*Figure S3.* An example problem from the textbook *Fundamentals of Physics*.

*Table S1.* Statistics of the close exam dataset. We report the number of problem instances in each exam and the ratio of problems in the exam that include detailed solutions. We further report the ratio of problems in different formats, including free-response, multiple-choice, and true-false. For reference, the number in parentheses denotes the grading points assigned to the problems.

|  | Data Mining | | Machine Learning | | Differential Equations | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Midterm | Final | Midterm | Final | Exam 1 | Exam 2 | Final |
| # Problems | 25 (90) | 24 (75) | 12 (56) | 16 (75) | 8 (100) | 8 (100) | 11 (95) |
| % Solutions | 56.0% (58) | 16.7% (19) | 100.0% (56) | 31.2% (26) | 100.0% (100) | 100.0% (100) | 90.9% (90) |
| % Free-response | 40.0% (46) | 33.3% (29) | 66.7% (38) | 81.3% (62) | 100.0% (100) | 100.0% (100) | 90.9% (90) |
| % Multiple-choice | 28.0% (28) | 29.2% (28) | 33.3% (18) | 18.7% (13) | 0.0% (0) | 0.0% (0) | 9.1% (5) |
| % True-false | 32.0% (16) | 37.5% (18) | 0.0% (0) | 0.0% (0) | 0.0% (0) | 0.0% (0) | 0.0% (0) |

*Table S2.* Experimental results in terms of total scores under zero-shot learning on the exam dataset. The best performing score is highlighted in **bold** and second-best is underlined.

| Model | Setting | Data Mining | | Machine Learning | | Differential Equations | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Midterm | Final | Midterm | Final | Exam 1 | Exam 2 | Final |
| LLaMA-2-7B | Zero | 24 / 90 | 14 / 75 | 6 / 56 | 6/ 75 | 5 / 100 | 0 / 100 | 0 / 95 |
|  | Zero+CoT | 18 / 90 | 14 / 75 | 2 / 56 | 10 / 75 | 10 / 100 | 0 / 100 | 10 / 95 |
| LLaMA-2-70B | Zero | 23 / 90 | 18 / 75 | 18 / 56 | 12 / 75 | 20 / 100 | 5 / 100 | 0 / 95 |
|  | Zero+CoT | 31 / 90 | 18 / 75 | 10 / 56 | 11/ 75 | 35 / 100 | 10 / 100 | 0 / 95 |
| Claude2 | Zero | 37 / 90 | 26 / 75 | 28 / 56 | 35 / 75 | 35 / 100 | 30 / 100 | 20 / 95 |
|  | Zero+CoT | 33 / 90 | 38 / 75 | 22 / 56 | **41** / 75 | 25 / 100 | 15 / 100 | 20 / 95 |
| GPT-3.5 | Zero | 44 / 90 | 39 / 75 | 16 / 56 | 32 / 75 | 0 / 100 | 45 / 100 | 15 / 95 |
|  | Zero+CoT | 38 / 90 | 33 / 75 | 32 / 56 | 37 / 75 | 28 / 100 | 30 / 100 | 10 / 95 |
| GPT-4 | Zero | 56 / 90 | **44** / 75 | 30 / 56 | 37 / 75 | 25 / 100 | **80** / 100 | **25** / 95 |
|  | Zero+CoT | **58** / 90 | 32 / 75 | **40** / 56 | 35 / 75 | **50** / 100 | 70 / 100 | 15 / 95 |

**Problem**

If the particles in a system all move together, the com moves with them-no trouble there. But what happens when they move in different directions with different accelerations? Here is an example. The three particles in the Figure are initially at rest. Each experiences an external force due to bodies outside the three-particle system. The directions are indicated, and the magnitudes are $F_1 = 6.0$ N, $F_2 = 12$ N, and $F_3 = 14$ N. What is the acceleration of the center of mass of the system?

**Image**
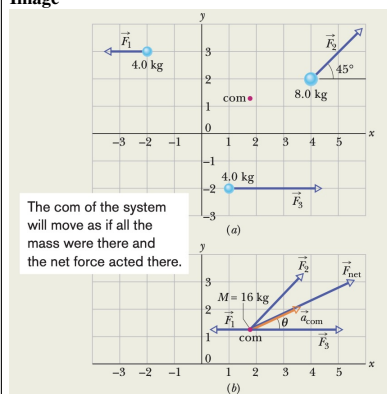


**Correct Solution**

The position of the center of mass is marked by a dot in the figure. We can treat the center of mass as if it were a real particle, with a mass equal to the system's total mass $M = 16$ kg. We can also treat the three external forces as if they act at the center of mass (Figure b). We can now apply Newton's second law $\left( \vec{F}_{\text{net}} = m\vec{a} \right)$ to the center of mass, writing

$$\vec{F}_{\text{net}} = M\vec{a}_{\text{com}}, \tag{S4}$$

$$\vec{F}_1 + \vec{F}_2 + \vec{F}_3 = M\vec{a}_{\text{com}}, \tag{S5}$$

$$\vec{a}_{\text{com}} = \frac{\vec{F}_1 + \vec{F}_2 + \vec{F}_3}{M}. \tag{S6}$$

The equation tells us that the acceleration $\vec{a}_{\text{com}}$ of the center of mass is in the same direction as the net external force $\vec{F}_{\text{net}}$ on the system (Figure b). Because the particles are initially at rest, the center of mass must also be at rest. As the center of mass then begins to accelerate, it must move off in the common direction of $\vec{a}_{\text{com}}$ and $\vec{F}_{\text{net}}$. We can evaluate the right side of Eq. S6 directly on a vector-capable calculator, or we can rewrite Eq. S6 in component form, find the components of $\vec{a}_{\text{com}}$, and then find $\vec{a}_{\text{com}}$. Along the $x$ axis, we have

$$a_{\text{com},x} = \frac{F_{1x} + F_{2x} + F_{3x}}{M} = \frac{-6.0 \text{ N} + (12 \text{ N}) \cos 45° + 14 \text{ N}}{16 \text{ kg}} = 1.03 \text{ m/s}^2. \tag{S7}$$

Along the $y$ axis, we have

$$a_{\text{com},y} = \frac{F_{1y} + F_{2y} + F_{3y}}{M} = \frac{0 + (12 \text{ N}) \sin 45° + 0}{16 \text{ kg}} = 0.530 \text{ m/s}^2. \tag{S8}$$

From these components, we find that $\vec{a}_{\text{com}}$ has the magnitude

$$a_{\text{com}} = \sqrt{(a_{\text{com},x})^2 + (a_{\text{com},y})^2} = 1.16 \text{ m/s}^2. \tag{S9}$$

**Final Answer:** $1.16$ m/s$^2$

*Figure S4.* The example from the textbook *Fundamentals of Physics*.

**Problem**

At time $t = 0$ a tank contains $Q_0$ lb of salt dissolved in 100 gal of water; see Figure 2.3.1. Assume that water containing $\frac{1}{4}$ lb of salt/gal is entering the tank at a rate of $r$ gal/min and that the well-stirred mixture is draining from the tank at the same rate. Set up the initial value problem that describes this flow process. By finding the amount of salt $Q(t)$ in the tank at any time, and the limiting amount $Q_L$ that is present after a very long time, if $r = 3$ and $Q_0 = 2Q_L$, find the time $T$ after which the salt level is within 2% of $Q_L$.
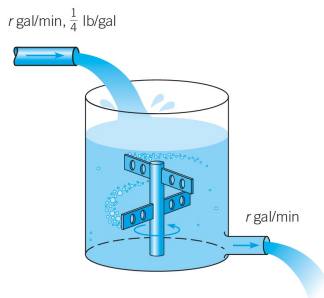
**Image**



$r$ gal/min, $\frac{1}{4}$ lb/gal

$r$ gal/min

**FIGURE 2.3.1**   The water tank in Example 1.

**Correct Solution**

We assume that salt is neither created nor destroyed in the tank. Therefore variations in the amount of salt are due solely to the flows in and out of the tank. More precisely, the rate of change of salt in the tank, $dQ/dt$, is equal to the rate at which salt is flowing in minus the rate at which it is flowing out. In symbols,

$$\frac{dQ}{dt} = \text{rate in} - \text{rate out}$$

The rate at which salt enters the tank is the concentration $\frac{1}{4}$ lb/gal times the flow rate $r$ gal/min, or $(r/4)$ lb/min. To find the rate at which salt leaves the tankl we need to multiply the concentration of salt in the tank by the rate of outflow, $r$ gal/min. Since the rates of flow in and out are equal, the volume of water in the tank remains constant at 100 gal, and since the mixture is "well-stirred," the concentration throughout the tank is the same, namely, $[Q(t)/100]$ lb/gal. Therefore the rate at which salt leaves the tank is $[rQ(t)/100]$ lb/min. Thus the differential equation governing this process is

$$\frac{dQ}{dt} = \frac{r}{4} - \frac{rQ}{100}$$

The initial condition is

$$Q(0) = Q_0$$

Upon thinking about the problem physically, we might anticipate that eventually the mixture originally in the tank will be essentially replaced by the mixture flowing in, whose concentration is $\frac{1}{4}$ lb/gal. Consequently, we might expect that ultimately the amount of salt in the tank would be very close to 25 lb. We can also find the limiting amount $Q_L = 25$ by setting $dQ/dt$ equal to zero and solving the resulting algebraic equation for $Q$. Rewriting it in the standard form for a linear equation, we have

$$\frac{dQ}{dt} + \frac{rQ}{100} = \frac{r}{4}$$

Thus the integrating factor is $e^{rt/100}$ and the general solution is

$$Q(t) = 25 + ce^{-rt/100}$$

where $c$ is an arbitrary constant. To satisfy the initial condition, we must choose $c = Q_0 - 25$. Therefore the solution of the initial value problem is

$$Q(t) = 25 + (Q_0 - 25)e^{-rt/100}$$

$$Q(t) = 25(1 - e^{-rt/100}) + Q_0 e^{-rt/100}$$

From above Equations, you can see that $Q(t) \to 25$ (lb) as $t \to \infty$, so the limiting value $Q_L$ is 25 , confirming our physical intuition. Further, $Q(t)$ approaches the limit more rapidly as $r$ increases. In interpreting the solution, note that the second term on the right side is the portion of the original salt that remains at time $t$, while the first term gives the amount of salt in the tank due to the action of the flow processes. Now suppose that $r = 3$ and $Q_0 = 2Q_L = 50$; then

$$Q(t) = 25 + 25e^{-0.03t}$$

Since 2% of 25 is 0.5 , we wish to find the time $T$ at which $Q(t)$ has the value 25.5. Substituting $t = T$ and $Q = 25.5$ and solving for $T$, we obtain

$$T = (\ln 50)/0.03 \cong 130.400766848(\text{ min}).$$

**Final Answer:** $(\ln 50)/0.03$

*Figure S5.* The example from the textbook *Elementary Differential Equations and Boundary Value Problems.*
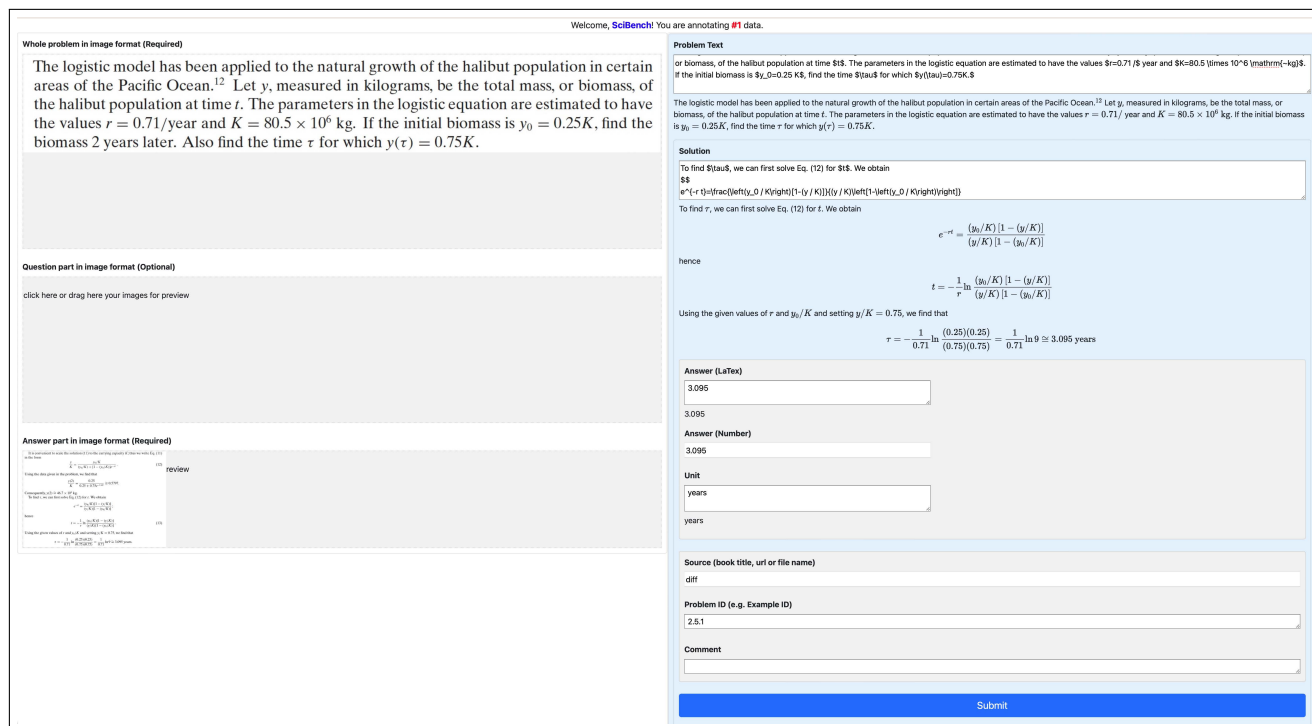
*Figure S6.* The UI design of data annotation.

## C. Experimental Details

### C.1. Prompts

The APIs of ChatGPT and GPT-4 have three message parameters: `SYSTEM`, `USER`, and `ASSISTANT`. The `SYSTEM` parameter represents the system prompt, which provides context and instructions to the model. The `USER` parameter is the training prompt or input provided by the user, and the `ASSISTANT` parameter contains the output of the model or the response. All system prompts and training prompts used in our experiments are provided below.

---

**System Prompt for Zero-Shot, Few-Shot, and Chain-of-Thought settings.**

Please provide a clear and step-by-step solution for a scientific problem in the categories of Chemistry, Physics, or Mathematics. The problem will specify the unit of measurement, which should not be included in the answer. Express the final answer as a decimal number with three digits after the decimal point. Conclude the answer by stating "The answer is therefore `\boxed{[ANSWER]}`."

---

**System Prompt for Few-Shot Learning + Python.**

Please provide a clear and step-by-step solution for a scientific problem in the categories of Chemistry, Physics, or Mathematics. The problem will specify the unit of measurement. Please translate the solution steps into Python code and encase the Python code within triple backticks for clarity.

---

**System Prompt for Few-Show Learning + Wolfram Language.**

Please provide a clear and step-by-step solution for a scientific problem in the categories of Chemistry, Physics, or Mathematics. The problem will specify the unit of measurement. Please translate the solution steps into Wolfram code and encase the Wolfram Language code within triple backticks for clarity.

---

**System Prompt for Evaluation Protocol.**

Examine the given problem, the correct solution, and the model's solution. Identify the reason for the error in the model's solution based on the following 10 categories:

1. Logical Decomposition and Analysis Skills: This ability involves decomposing the problem into smaller, manageable parts, and understanding the relationships between these parts.

2. Identification of Assumptions: This skill involves the AI's ability to recognize relevant and necessary assumptions in the problem.

3. Spatial Perception: This is important for understanding problems in areas such as physics and chemistry, where you need to visualize molecules, forces, fields, etc.

4. Causal Reasoning: This is the ability to understand cause and effect relationships.

5. Problem Deduction Skills: This pertains to the ability to infer and deduce potential solutions or underlying principles from the given information in a problem.

6. Abstract Reasoning: This skill involves the ability to understand complex concepts that can't be perceived physically, and to recognize patterns or relationships beyond concrete examples.

7. Scientific Literacy: This skill involves a comprehensive understanding of key scientific principles, terminology, and methodologies across a range of disciplines.

8. Code Conversion Skills: This denotes the ability to accurately translate solution steps into different programming languages, like Python or Wolfram, without syntax errors.

9. Logical Reasoning: This is the ability to make a reasoned argument and to identify fallacies or inconsistencies in an argument or set of data.

10. Calculation Skills: This involves the ability to accurately carry out mathematical operations and computations.

Conclude your final error reason category number within \boxed{}.

---

**Training Prompt for Zero-Shot Chain-of-Thought.**

*Stage 1:*

Input: [Input-Question] Let's think step by step.

Output: <explanation>

*Stage 2:*

Input: [Input-Question] Let's think step by step. [Explanation]. Therefore, the answer is:

Output: <answer>

---

**Training Prompt for Few-Shot Chain-of-Thought.**

Input:

Problem 1: [Question 1] Explanation for Problem 1: [Explanation 1]. The answer is \boxed{[Answer 1]}.

Problem 2: [Question 2] Explanation for Problem 2: [Explanation 2]. The answer is \boxed{[Answer 2]}.

...

Problem n: [Question n] Explanation for Problem n: [Explanation n]. The answer is \boxed{[Answer n]}.

Problem n+1: [Question n+1]

Output: Explanation for Problem n+1: <explanation>. The answer is \boxed{<answer>}.

---

**Training Prompt for Few-Shot Python or Wolfram Language.**

Input:

Problem 1: [Question 1] Explanation for Problem 1: [Explanation 1]. Python/Wolfram language for Problem 1: ```[Python/Wolfram code 1]```.

Problem 2: [Question 2] Explanation for Problem 2: [Explanation 2]. Python/Wolfram language for Problem 2: ```[Python/Wolfram code 2]```.

...

Problem n: [Question n] Explanation for Problem n: [Explanation n]. Python/Wolfram language for Problem n: ```[Python/Wolfram code n]```.

Problem n+1: [Question n+1]

Output: Explanation for Problem n+1: <explanation>. Python/Wolfram language for Problem n+1: ```[Python/Wolfram code n+1]```.

---

**Training Prompt for Evaluation Protocol.**

Input: The question is [input-question]. The correct solution is [Correct-Solution]. The model solution is [Model-Solution].

Output: <Error Type>

---

**Training Prompt for Evaluation Protocol in Python or Wolfram Language.**

Input: The question is [input-question]. The correct solution is [Correct-Solution]. The model solution is [Model-Solution]. The translated program generates the answer as [Program Generated Answer], which is treated as model's output answer.

Output: <Error Type>

---

## C.2. Implementation Details

All model output is extracted using \boxed{} notation. To prevent any missed extractions, we supplement this process with a manual check. For both Python and Wolfram settings, we extract the programming language with the triple backtick ```, subsequently executing it within the corresponding language. The entirety of our code can be accessed via this repository.

## C.3. Additional Experiment on Wolfram Language

The experiment results and error analysis for using Wolfram Language as external tools are presented in Figure S7 and Figure S8, compared with using CoT and Python Language. We observe that the use of external tools can weaken other
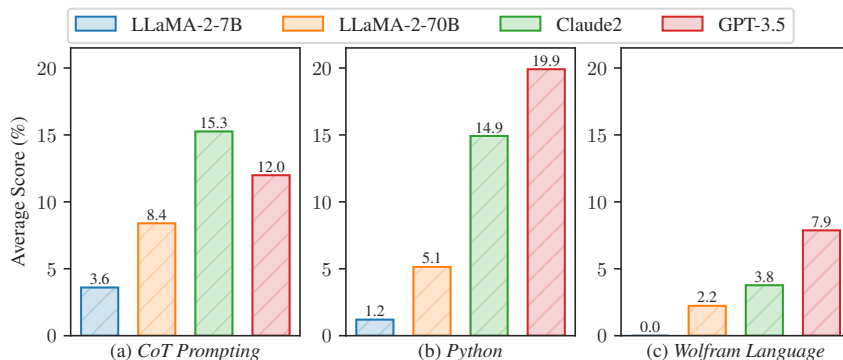


*Figure S7.* Comparison between few-shot learning with external tools.

skills, particularly the code conversion skills. This issue becomes particularly prominent when using the Wolfram Language, with 46.9% error rate in code conversion skill. Despite providing grammar specifications in system prompts and a few examples as demonstrations, most attempts of code conversion result in syntax errors. In Wolfram Language, the error mainly comes from the violation of variable rules (for instance, Wolfram Language reserves certain letters such as $E$ as protected symbols and disallows underscores in variable names) or incorrect usage of certain functions. This observation suggests a potential improvement for LLM when using Wolfram Language.

# D. Problem Solving Abilities of Current LLMs

## D.1. Assessment of the Evaluation Protocol

In order to assess the effectiveness of our evaluation protocol's classification, we enlisted the assistance of two annotators to determine whether the errors identified by the model verifier were accurate or not. Through the annotation of 151 samples across different settings, we observed that 123 of them were correctly classified, resulting in an accuracy rate of 81.45%. Two human annotators participate in the process. Decisions on the final abilities are determined by annotators, aided by assistants. By going through errors, these two annotators develop ten abilities and then employ a Language Learning Model (LLM) as a third evaluator to suggest additional abilities. They then compare and refine their findings based on this input. Ultimately, the final outcomes are determined by the annotators. After LLM annotate the error reasons, we conduct human-check by sampling 151 examples across all settings to make sure the annotations make sense. We make this human-AI cooperated analysis pipeline to reduce the cost of human post-analysis, while incorporate human checking to make sure the correctness of LLM decision and try to reduce the risk that reviewer mentioned. Though not perfect, we believe it can be another type of analyzing framework for future study of LLM problem-solving.

## D.2. Examples

In the context of each specific capability, we present several exemplary errors accompanied by their corresponding classifications and explanations derived from the GPT model. Referencing Figure S9, the ChatGPT solution employing the Chain-of-Thought method corresponds to error category 4, "Causal Reasoning". The model explanation posits that "The error reason category is 4. Causal Reasoning. The model solution uses the relativistic momentum formula to calculate the momentum of the electron, which is not necessary for this problem since the electron is traveling at only $1.00\%$ of the speed of light. The relativistic momentum formula is only needed when the velocity of the particle is a significant fraction of the speed of light. Therefore, using this formula in this problem leads to an incorrect value for the momentum and consequently an incorrect value for the de Broglie wavelength." Meanwhile, the ChatGPT solution that doesn't utilize the Chain-of-Thought method is categorized under error case 10, "Calculation Skills". This example is further highlighted as an illustrative showcase for explanations of the Chain-of-Thought method in Section 5. More examples resulting in different error types are shown in Figures S10 to S19.
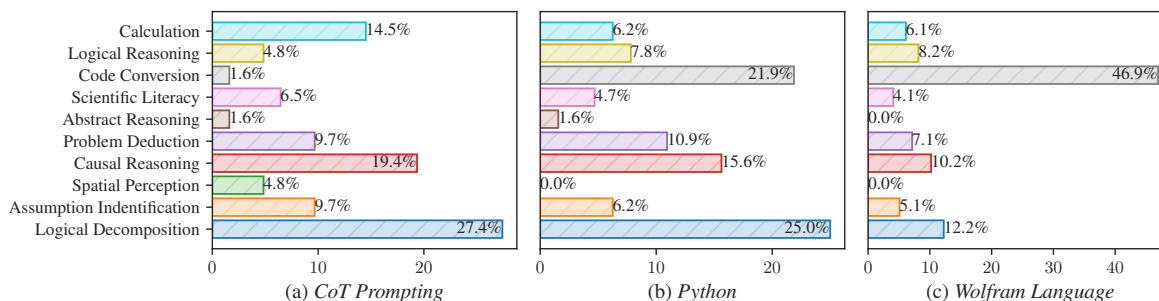


*Figure S8.* Error Analysis between few-shot learning with external tools.

**Problem**

Calculate the de Broglie wavelength of an electron traveling at $1.00\%$ of the speed of light.

**Correct Solution**

The mass of an electron is $9.109 \times 10^{-31}$ kg. One percent of the speed of light is

$$v = (0.0100)\left(2.998 \times 10^8 \text{ m} \cdot \text{s}^{-1}\right) = 2.998 \times 10^6 \text{ m} \cdot \text{s}^{-1}$$

The momentum of the electron is given by

$$p = m_{\text{e}}v = \left(9.109 \times 10^{-31} \text{ kg}\right)\left(2.998 \times 10^6 \text{ m} \cdot \text{s}^{-1}\right) = 2.73 \times 10^{-24} \text{ kg} \cdot \text{m} \cdot \text{s}^{-1}$$

The de Broglie wavelength of this electron is

$$\lambda = \frac{h}{p} = \frac{6.626 \times 10^{-34} \text{ J} \cdot \text{s}}{2.73 \times 10^{-24} \text{ kg} \cdot \text{m} \cdot \text{s}^{-1}} = 2.43 \times 10^{-10} \text{ m} = 243\text{pm}$$

This wavelength is of atomic dimensions.

**Final Answer:** 243pm

**Chatgpt Solution with Chain-of-Thought Method**

Let's start by recalling the de Broglie wavelength formula:$\lambda = \frac{h}{p}$ where $\lambda$ is the de Broglie wavelength, $h$ is Planck's constant, and $p$ is the momentum of the particle. To calculate the momentum of the electron, we can use the relativistic momentum formula:

$$\textcolor{red}{p = \frac{m_e v}{\sqrt{1 - \frac{v^2}{c^2}}}}$$

where $m_e$ is the mass of the electron, $v$ is its velocity, and $c$ is the speed of light. Plugging in the given values, we get:

$$p = \frac{(9.11 \times 10^{-31}\text{kg})(0.01c)}{\sqrt{1 - (0.01c)^2/c^2}} = 8.19 \times 10^{-23}\text{kg} \cdot \text{m/s}$$

Now we can plug this momentum into the de Broglie wavelength formula:

$$\lambda = \frac{h}{p} = \frac{6.626 \times 10^{-34}\text{J} \cdot \text{s}}{8.19 \times 10^{-23}\text{kg} \cdot \text{m/s}} = 8.09\text{pm}$$

Therefore, the de Broglie wavelength of the electron is $\boxed{8.09}$.

**Final Answer:** 8.09pm

**ChatGPT Solution without Chain-of-Thought Method**

The de Broglie wavelength of a particle is given by the equation:$\lambda = \frac{h}{p}$ where $h$ is the Planck constant and $p$ is the momentum of the particle. The momentum of an electron is given by:

$$p = mv$$

where $m$ is the mass of the electron and $v$ is its velocity. We are given that the electron is traveling at $1.00\%$ of the speed of light, which is:$v = 0.01c$ where $c$ is the speed of light. The mass of an electron is:$m = 9.11 \times 10^{-31}$, kg Putting it all together, we have:

$$\lambda = \frac{h}{mv} = \frac{h}{m(0.01c)}$$

Substituting the values of $h$, $m$, and $c$, we get:

$$\lambda = \frac{6.626 \times 10^{-34}, \text{J} \cdot \text{s}}{9.11 \times 10^{-31}, \text{kg} \cdot 0.01 \cdot 2.998 \times 10^8, \text{m/s}} = \textcolor{red}{3.31 \times 10^{-12}\text{m} = 3.31\text{pm}}$$

Therefore, the de Broglie wavelength of the electron is $\boxed{3.31\text{pm}}$

**Final Answer:** 3.31pm

*Figure S9.* An example problem, inaccurately solved by ChatGPT with and without the Chain of Thought method, is provided to highlight different types of errors. In the example, the mistaken step is highlighted in red.

**Problem**

Find the length of the cardioid $r = 1 + \sin\theta$.

**Correct Solution**

The cardioid's full length is given by the parameter interval $0 \le \theta \le 2\pi$, so

$$L = \int_0^{2\pi} \sqrt{r^2 + \left(\frac{dr}{d\theta}\right)^2}\, d\theta = \int_0^{2\pi} \sqrt{(1+\sin\theta)^2 + \cos^2\theta}\, d\theta \quad = \int_0^{2\pi} \sqrt{2 + 2\sin\theta}\, d\theta$$

We could evaluate this integral by multiplying and dividing the integrand by $\sqrt{2 - 2\sin\theta}$, or we could use a computer algebra system. In any event, we find that the length of the cardioid is $L = 8$.

**Final Answer:** 8

**Model Solution**

To find the length of the cardioid, we use the formula for arc length in polar coordinates:

$$L = \int_\alpha^\beta \sqrt{r^2 + \left(\frac{dr}{d\theta}\right)^2}\, , d\theta,$$

where $\alpha$ and $\beta$ are the angles that correspond to one full loop of the cardioid (i.e. $\alpha = 0$ and $\beta = 2\pi$). First, we find $\frac{dr}{d\theta}$:

$$\frac{dr}{d\theta} = \frac{d}{d\theta}(1 + \sin\theta) = \cos\theta.$$

Then, we plug in $r = 1 + \sin\theta$ and $\frac{dr}{d\theta} = \cos\theta$ into the formula for arc length:

$$L = \int_0^{2\pi} \sqrt{(1+\sin\theta)^2 + \cos^2\theta}$$
$$d\theta = \int_0^{2\pi} \sqrt{1 + 2\sin\theta + \sin^2\theta + \cos^2\theta}$$
$$d\theta = \int_0^{2\pi} \sqrt{2 + 2\sin\theta}$$
$$d\theta = 4\int_0^{\pi} \sqrt{2 + 2\sin\theta}$$
$$d\theta \quad \text{(due to symmetry)} = 8\int_0^{\pi/2} \sqrt{2 + 2\sin\theta}$$
$$d\theta = 8\int_0^{\pi/2} \sqrt{4\cos^2 \frac{\theta}{2}}$$
$$d\theta \quad \text{(using the half-angle formula)} = 16\int_0^{\pi/2} \cos\frac{\theta}{2}$$
$$d\theta = 32\sin\frac{\theta}{2}\Big|_0^{\pi/2}$$
$$= 32.$$

Therefore, the length of the cardioid is 32, and the unit of the answer is whatever unit was used for $r$ (e.g. meters, centimeters, etc.).

**Final Answer:** 32

**Error Reason**

The model solution made an error in the calculation of the integral $\int_0^{2\pi} \sqrt{2 + 2\sin\theta}\, d\theta$. The correct solution correctly simplified this integral to $4\int_0^{\pi} \sqrt{2 + 2\sin\theta}\, d\theta$ due to the symmetry of the integrand, and then used a substitution to evaluate the integral. The model solution, on the other hand, incorrectly evaluated this integral as $8\pi$.

This error falls under the category of $\boxed{\text{10. Calculation Skills}}$.

*Figure S10.* An example problem is inaccurately solved by error reason 10. Calculation Skills. The mistaken step is highlighted in red.

**Problem**
Use Simpson's Rule with $n = 10$ to approximate $\int_1^2 (1/x)\,dx$.

**Correct Solution**
Putting $f(x) = 1/x$, $n = 10$, and $\Delta x = 0.1$ in Simpson's Rule, we obtain

$$\int_1^2 \frac{1}{x}\,dx$$
$$\approx S_{10}$$
$$= \frac{\Delta x}{3}[f(1) + 4f(1.1) + 2f(1.2) + 4f(1.3) + \cdots + 2f(1.8) + 4f(1.9) + f(2)]$$
$$= \frac{0.1}{3}\left(\frac{1}{1} + \frac{4}{1.1} + \frac{2}{1.2} + \frac{4}{1.3} + \frac{2}{1.4} + \frac{4}{1.5} + \frac{2}{1.6} + \frac{4}{1.7} + \frac{2}{1.8} + \frac{4}{1.9} + \frac{1}{2}\right)$$
$$\approx 0.693150$$

**Final Answer:** 0.693150

**Model Solution**
Explanation for Problem 6: With $n = 10$ we have $h = \frac{2-1}{10} = \frac{1}{10}$ and $x_i = 1 + ih$ for $i = 0, 1, 2, \ldots, 10$. Then Simpson's Rule gives

$$\int_1^2 \frac{1}{x}\,dx \approx \frac{h}{3}[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + \ldots + 2f(x_8) + 4f(x_9) + f(x_{10})]$$

where $f(x) = \frac{1}{x}$.
wolfram language for Problem 6:
(* Constants *)
n = 10;
a = 1;
b = 2;
(* Function to integrate *)
f[x_] := 1/x;
(* Simpson's Rule *)
h = (b - a)/n;
x = Table[a + i*h, i, 0, n];
approximation = (h/3)*Total[Table[If[EvenQ[i], 2, 4]*f[x[[i]]], i, 1, n - 1] + f[x[[0]]], f[x[[n]]], 4*f[x[[n/2]]]]];
approximation
**Final Answer:** "Objects of unequal length in {4,20/11,10/3,20/13,20/7,4/3,5/2,20/17,20/9}+{1/List,10/19,20/7} cannot be combined."

**Error Reason**
The error reason category for the model's solution is  8. Code Conversion Skills . The model's solution contains a syntax error in the Wolfram Language code, which causes the program to exceed the recursion limit and terminate prematurely. This error likely occurred during the translation of the correct solution's code into Wolfram Language.

*Figure S11.* An example problem is inaccurately solved by error reason 8. Code Conversion Skills. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes. In the example, the mistaken step is highlighted in red.

**Problem**
Use Simpson's Rule with $n = 10$ to approximate $\int_1^2 (1/x)\,dx$.

**Correct Solution**
The equations for $c_1$ and $c_2$ are
$$c_1(\alpha - E) + c_2\beta = 0 \quad \text{and} \quad c_1\beta + c_2(\alpha - E) = 0$$

For $E = \alpha + \beta$, either equation yields $c_1 = c_2$. Thus, $\psi_b = c_1(2p_{z1} + 2p_{z2})$. The value of $c_1$ can be found by requiring that the wave function be normalized. The normalization condition on $\psi_\pi$ gives $c_1^2(1 + 2S + 1) = 1$. Using the Hückel assumption that $S = 0$, we find that $c_1 = 1/\sqrt{2}$. Substituting $E = \alpha - \beta$ into either of the equations for $c_1$ and $c_2$ yields $c_1 = -c_2$, or
$$\psi_a = c_1(2p_{z1} - 2p_{z2})$$

The normalization condition gives $c^2(1 - 2S + 1) = 1$, or $c_1 = 1/\sqrt{2}$.
**Final Answer:** 0.7071067

**Model Solution**
therefore not provided as this problem requires visual representations and cannot be solved through text alone.

**Error Reason**
The error reason category is  3. Spatial Perception . This problem requires visual representations to find the bonding and antibonding Hückel molecular orbitals for ethene. The model's solution is incomplete as it only provides equations and does not provide any visual representation.

*Figure S12.* An example problem is inaccurately solved by error reason 3. Spatial Perception. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes.

---

**Problem**
What is the mean speed, $\bar{c}$, of $N_2$ molecules in air at $25^\circ$ C

**Correct Solution**
The integral required is

$$\bar{c} = 4\pi \left( \frac{M}{2\pi RT} \right)^{3/2} \int_0^\infty v^3 e^{-Mv^2/2RT} \, dv \quad = 4\pi \left( \frac{M}{2\pi RT} \right)^{3/2} \times \frac{1}{2} \left( \frac{2RT}{M} \right)^2 = \left( \frac{8RT}{\pi M} \right)^{1/2}$$

where we have used the standard result from tables of integrals (or software) that

$$\int_0^\infty x^3 e^{-ax^2} \, dx = \frac{1}{2a^2}$$

Substitution of the data then gives

$$\bar{c} = \left( \frac{8 \times \left( 8.3141 \text{ J K}^{-1} \text{ mol}^{-1} \right) \times (298 \text{ K})}{\pi \times \left( 28.02 \times 10^{-3} \text{ kg mol}^{-1} \right)} \right)^{1/2} = 475 \text{ m s}^{-1}$$

where we have used 1 J = 1 kg m$^2$ s$^{-2}$.
**Final Answer:** 475 m s$^{-1}$

**Model Solution**
To calculate this, we can use the formula for the root-mean-square (rms) speed of gas molecules:

$$c_{\text{rms}} = \sqrt{\frac{3kT}{m}}$$

where $k$ is the Boltzmann constant, $T$ is the temperature in Kelvin, and $m$ is the mass of the molecule. For $N_2$, $m = 28$ amu. Plugging in the values, we get:

$$c_{\text{rms}} = \sqrt{\frac{3 \times 1.38 \times 10^{-23} \text{ J/K} \times (25 + 273) \text{ K}}{28 \times 1.66 \times 10^{-27} \text{ kg/amu}}} \approx 515 \text{ m s}^{-1}$$

Note that we converted the temperature from Celsius to Kelvin and the mass from atomic mass units (amu) to kilograms.
**Final Answer:** 515 m s$^{-1}$

**Error Reason**
The error reason category is $\boxed{\text{7. Scientific Literacy}}$. The model solution uses the formula for the root-mean-square (rms) speed of gas molecules, which is only valid for ideal gases. However, the given problem does not specify that the gas is ideal, and in fact, air is not an ideal gas. The correct solution uses a more general formula that applies to all gases, including non-ideal gases like air. Therefore, the model's solution is not applicable to the given problem.

*Figure S13.* An example problem is inaccurately solved by error reason 7. Scientific Literacy. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes. In the example, the mistaken step is highlighted in red, while the correct step is highlighted in blue.

---

**Problem**
What is the number of possible four-letter code words, selecting from the 26 letters in the alphabet?

**Correct Solution**
The number of possible four-letter code words, selecting from the 26 letters in the alphabet, in which all four letters are different is

$$_{26}P_4 = (26)(25)(24)(23) = \frac{26!}{22!} = 358,800.$$

**Final Answer:** 358800

**Model Solution**
The number of possible four-letter code words, selecting from the 26 letters in the alphabet is

$$26^4 = 456,976.$$

**Final Answer:** 456976

**Error Reason**
The model's error reason category is $\boxed{\text{9. Logical Reasoning}}$. The model did not consider the condition that all four letters should be different. The correct solution considers this condition and uses the permutation formula to calculate the number of possible four-letter code words.

*Figure S14.* An example problem is inaccurately solved by error reason 9. Logical reasoning. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes. In the example, the mistaken step is highlighted in red.

---

**Problem**
Find the bonding and antibonding Hückel molecular orbitals for ethene.

---

**Correct Solution**
The equations for $c_1$ and $c_2$ associated are
$$c_1(\alpha - E) + c_2\beta = 0 \quad \text{and} \quad c_1\beta + c_2(\alpha - E) = 0$$

For $E = \alpha + \beta$, either equation yields $c_1 = c_2$. Thus,
$$\psi_b = c_1(2p_{z1} + 2p_{z2})$$

The value of $c_1$ can be found by requiring that the wave function be normalized. The normalization condition on $\psi_\pi$ gives $c_1^2(1 + 2S + 1) = 1$. Using the Hückel assumption that $S = 0$, we find that $c_1 = 1/\sqrt{2}$. Substituting $E = \alpha - \beta$ into either of the equations for $c_1$ and $c_2$ yields $c_1 = -c_2$, or
$$\psi_a = c_1(2p_{z1} - 2p_{z2})$$

The normalization condition gives $c^2(1 - 2S + 1) = 1$, or $c_1 = 1/\sqrt{2}$.
**Final Answer:** 0.70710678

---

**Model Solution**
Two bonding and two antibonding Hückel molecular orbitals for ethene. The unit of the answer is not applicable as it is a qualitative description of the molecular orbitals.
**Final Answer:** None

---

**Error Reason**
The error reason category for the model's solution is $\boxed{\text{6. Abstract Reasoning}}$. The model's solution is incorrect because it did not consider the Hückel molecular orbital theory, which predicts that there are only two pi molecular orbitals for ethene: one bonding and one antibonding. This is due to the fact that ethene has only two pi electrons, which can occupy the two pi molecular orbitals. Therefore, the model's solution of two bonding and two antibonding molecular orbitals is not possible for ethene.

*Figure S15.* An example problem is inaccurately solved by error reason 6. Abstract Reasoning. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes.

---

**Problem**
A living room has floor dimensions of 3.5 m and 4.2 m and a height of 2.4 m. What does the air in the room weigh when the air pressure is 1.0 atm?

---

**Correct Solution**
The air's weight is equal to $mg$, where $m$ is its mass Mass $m$ is related to the air density $\rho$ and the air volume $V$ ($\rho = m/V$).
Calculation: Putting the two ideas together and taking the density of air at 1.0 atm, we find

$$mg = (\rho V)g \quad = \left(1.21\text{ kg/m}^3\right)(3.5\text{ m} \times 4.2\text{ m} \times 2.4\text{ m})\left(9.8\text{ m/s}^2\right) = 418\text{ N}$$

**Final Answer:** 418 N

---

**Model Solution**
To find the weight of the air in the room, we need to first calculate the volume of the room: $V = lwh = 3.5\text{ m} \times 4.2\text{ m} \times 2.4\text{ m} = 35.28\text{ m}^3$. Next, we can use the ideal gas law to find the mass of the air in the room: $PV = nRT$ where $P$ is the pressure, $V$ is the volume, $n$ is the number of moles of gas, $R$ is the gas constant, and $T$ is the temperature. Assuming room temperature of $20°$C (293 K), we can solve for $n$:

$$n = \frac{PV}{RT} = \frac{(1.0\text{ atm})(35.28\text{ m}^3)}{(0.0821\text{ L atm/mol K})(293\text{ K})} = 1,610\text{ mol}$$

Finally, we can find the weight of the air using the molar mass of air (approximately 28.97 g/mol):

$$W = mg = nMg = (1,610\text{ mol})(0.02897\text{ kg/mol})(9.81\text{ m/s}^2) = 101,640\text{ N}$$

**Final Answer:** 101640 N

---

**Error Reason**
The error in the model's solution is that they used the ideal gas law to find the mass of the air in the room, which assumes that the air behaves like an ideal gas. However, the ideal gas law is not applicable in this case because the problem does not provide information about the temperature of the air. The correct approach is to use the density of air at 1.0 atm and the volume of the room to find the mass of the air, as shown in the correct solution. Therefore, the reason for the error in the model's solution is $\boxed{\text{2. Identification of Assumptions}}$.

*Figure S16.* An example problem is inaccurately solved by error reason 2. Identification of Assumptions. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes. In the example, the mistaken step is highlighted in red.

---

**Problem**

From an ordinary deck of playing cards, cards are to be drawn successively at random and without replacement. What is the probability that the third spade appears on the sixth draw?

---

**Correct Solution**

Let $A$ be the event of two spades in the first five cards drawn, and let $B$ be the event of a spade on the sixth draw. Thus, the probability that we wish to compute is $P(A \cap B)$. It is reasonable to take

$$P(A) = \frac{\left(\begin{array}{c} 132 \end{array}\right)\left(\begin{array}{c} 39 \\ 3 \end{array}\right)}{\left(\begin{array}{c} 525 \end{array}\right)} = 0.274 \quad \text{and} \quad P(B \mid A) = \frac{11}{47} = 0.234$$

The desired probability, $P(A \cap B)$, is the product of those numbers: $P(A \cap B) = (0.274)(0.234) = 0.064$

**Final Answer:** 0.064

---

**Model Solution**

The probability that the third spade appears on the sixth draw is

$$\frac{{}_{13}C_2 \cdot {}_{39}C_4}{{}_{52}C_6} = \frac{78 \cdot 741,948}{20,358,520} = \frac{1,441}{5,577}.$$

**Final Answer:** 0.258

---

**Error Reason**

The model's solution is incorrect because it calculates the probability of getting two spades in the first five cards and any spade on the sixth draw, instead of specifically calculating the probability of the third spade appearing on the sixth draw given that two spades have already appeared in the first five draws. Therefore, the error reason category is $\boxed{\text{4. Causal Reasoning}}$

*Figure S17.* An example problem is inaccurately solved by error reason 4. Causal Reasoning. The mistaken step is highlighted in red.

---

**Problem**

Gas cylinders of $CO_2$ are sold in terms of weight of $CO_2$. A cylinder contains 50lb (22.7 kg ) of $CO_2$. How long can this cylinder be used in an experiment that requires flowing $CO_2$ at 293 K $(\eta = 146\mu P)$ through a 1.00-m-long tube (diameter $= 0.75$ mm) with an input pressure of $1.05$ atm and output pressure of $1.00$ atm ? The flow is measured at the tube output

---

**Correct Solution**

The gas flow rate $\Delta V / \Delta t$ is

$$\frac{\Delta V}{\Delta t} = \frac{\pi r^4}{16\eta L P_0}\left(P_2^2 - P_1^2\right) = \quad 2.76 \times 10^{-6}\ \mathrm{m}^3\ \mathrm{s}^{-1}$$

Converting the $CO_2$ contained in the cylinder to the volume occupied at 298 K and 1 atm pressure, we get

$$n_{CO_2} = 22.7\ \mathrm{kg}\left(\frac{1}{0.044\ \mathrm{kg\ mol}^{-1}}\right) = 516\ \mathrm{mol},\ V = \frac{nRT}{P} = 1.24 \times 10^4\ \mathrm{L}\left(\frac{10^{-3}\ \mathrm{m}^3}{\mathrm{L}}\right) = 12.4\ \mathrm{m}^3$$

Given the effective volume of $CO_2$ contained in the cylinder, the duration over which the cylinder can be used is $\frac{12.4\ \mathrm{m}^3}{2.76 \times 10^{-6}\ \mathrm{m}^3\ \mathrm{s}^{-1}} = 4.49 \times 10^6\ \mathrm{s}$. This time corresponds to roughly 52 days.

**Final Answer:** 52 days

---

**Model Solution**

The volume of $CO_2$ in the cylinder is given by

$$V = \frac{m}{\rho} = \frac{22.7\ \mathrm{kg}}{1.98\ \mathrm{kg\ m}^{-3}} = 11.5\ \mathrm{m}^3$$

The volume of $CO_2$ that flows through the tube is given by

$$V = \frac{\pi}{4}d^2 L = \frac{\pi}{4}\left(0.75 \times 10^{-3}\ \mathrm{m}\right)^2 (1.00\ \mathrm{m}) = 1.77 \times 10^{-7}\ \mathrm{m}^3$$

The time required to flow $V$ volume of $CO_2$ is given by $t = \frac{V}{Q} = \frac{V P_2}{Q_m P_1} = \frac{\left(1.77 \times 10^{-7}\ \mathrm{m}^3\right)(1.00\ \mathrm{atm})}{(1.05\ \mathrm{atm} - 1.00\ \mathrm{atm})(146\mu P)} = 2.08 \times 10^6\ \mathrm{s}$. The number of seconds in a year is $3.15 \times 10^7\ \mathrm{s}$. Therefore, the cylinder can be used for $\frac{2.08 \times 10^6\ \mathrm{s}}{3.15 \times 10^7\ \mathrm{s\ yr}^{-1}} = 0.066\ \mathrm{yr} = 24.1\ \mathrm{days}$

**Final Answer:** 24.1 days

---

**Error Reason**

The model's solution is incorrect because it calculates the time required to flow a volume of $CO_2$ through the tube, but it does not take into account the total volume of $CO_2$ in the cylinder. The correct solution calculates the total volume of $CO_2$ in the cylinder and divides it by the flow rate to determine the duration over which the cylinder can be used. Therefore, the error reason category is $\boxed{\text{5. Problem Deduction Skills}}$.

*Figure S18.* An example problem is inaccurately solved by error reason 5. Problem Deduction Skills. The mistaken step is highlighted in red.

**Problem**

An astronaut whose height $h$ is 1.70 m floats "feet down" in an orbiting space shuttle at distance $r = 6.77 \times 10^6$ m away from the center of Earth. What is the difference between the gravitational acceleration at her feet and at her head?

**Correct Solution**

We can approximate Earth as a uniform sphere of mass $M_E$. Then, the gravitational acceleration at any distance $r$ from the center of Earth is

$$a_g = \frac{GM_E}{r^2}$$

We might simply apply this equation twice, first with $r = 6.77 \times 10^6$ m for the location of the feet and then with $r = 6.77 \times 10^6$ m $+ 1.70$ m for the location of the head. However, a calculator may give us the same value for $a_g$ twice, and thus a difference of zero, because $h$ is so much smaller than $r$. Here's a more promising approach: Because we have a differential change $dr$ in $r$ between the astronaut's feet and head, we should differentiate the equation with respect to $r$.

Calculations: The differentiation gives us

$$da_g = -2\frac{GM_E}{r^3}dr$$

where $da_g$ is the differential change in the gravitational acceleration due to the differential change $dr$ in $r$. For the astronaut, $dr = h$ and $r = 6.77 \times 10^6$ m. Substituting data into the equation, we find

$$da_g = -2\frac{\left(6.67 \times 10^{-11}\ \text{m}^3/\text{kg} \cdot \text{s}^2\right)\left(5.98 \times 10^{24}\ \text{kg}\right)}{(6.77 \times 10^6\ \text{m})^3}(1.70\ \text{m}) \quad = -4.37 \times 10^{-6}\ \text{m/s}^2$$

**Final Answer:** $-4.37 \times 10^{-6}\ \text{m/s}^2$

**Model Solution**

We can use the equation for the gravitational acceleration due to a point mass, given by Eq ($g = GM/r^2$), to find the gravitational acceleration at the astronaut's feet and head. Because the astronaut is in free fall, the difference between these two accelerations is the tidal acceleration, given by Eq ($\Delta g = 2GMh/r^3$).

Calculations: The gravitational acceleration at the astronaut's feet is

$$g_{\text{feet}} = \frac{GM}{r^2} = \frac{\left(6.67 \times 10^{-11}\ \text{N} \cdot \text{m}^2/\text{kg}^2\right)\left(5.98 \times 10^{24}\ \text{kg}\right)}{(6.77 \times 10^6\ \text{m})^2} = 9.78\ \text{m/s}^2.$$

The gravitational acceleration at the astronaut's head is

$$g_{\text{head}} = \frac{\textcolor{red}{GM}}{\textcolor{red}{(r+h)^2}} = \frac{\left(6.67 \times 10^{-11}\ \text{N} \cdot \text{m}^2/\text{kg}^2\right)\left(5.98 \times 10^{24}\ \text{kg}\right)}{(6.77 \times 10^6\ \text{m} + 1.70\ \text{m})^2} = 9.77\ \text{m/s}^2.$$

The difference between these two accelerations is

$$\Delta g = g_{\text{feet}} - g_{\text{head}} \quad = 9.78\ \text{m/s}^2 - 9.77\ \text{m/s}^2 = 0.01\ \text{m/s}^2 \quad = 1.0 \times 10^{-5}\ \text{m/s}^2.$$

**Final Answer:** $1.0 \times 10^{-5}\ \text{m/s}^2$

**Error Reason**

The model's solution is incorrect because it uses the equation for the gravitational acceleration due to a point mass, which is not applicable in this case since the Earth is not a point mass. The correct equation to use is the one given in the correct solution, which takes into account the fact that the Earth is a uniform sphere of mass. Therefore, the error reason category is | 1. Logical Decomposition and Analysis Skills |.

*Figure S19.* An example problem is inaccurately solved by error reason 1. Logical Decomposition and Analysis Skills. "Error Reason" denotes the output from the LLM Verifier utilized in the classification of error causes. In the example, the mistaken step is highlighted in red.