



# インサイト と GPT/LLM

株式会社インサイト

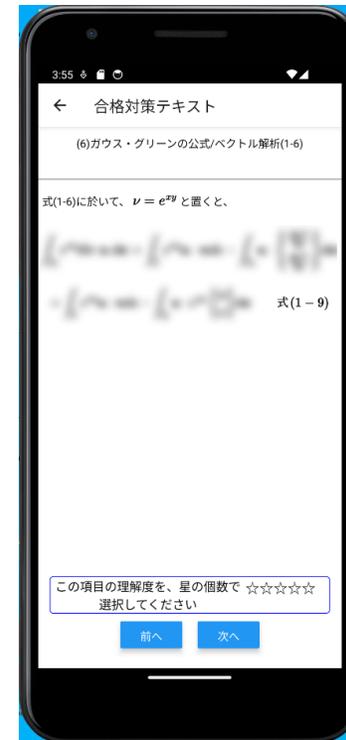
2023/11/17

# 目次

- スマホ教育アプリ開発 / 販売
- 対話型AI
- LLM(Large Language Model)とLLM迄の変遷
- LLaMA 2
- ELYZA-japanese-Llama-2-7b
- 弊社事例I: ELYZA-japanese-Llama-2-7b
- 教育アプリのコンテンツによるLLMチューニング

# スマホ教育アプリの開発 / 販売

- 計算力学技術者認定試験2級（固体力学） Google Play Store/ App Store ➤ 「CAE 固体」で検索



# スマホ教育アプリの開発 / 販売

- 隙間時間に学習できるアプリを目指す
- 2023年 Android 及び iOS をリリース
- 作成時使用したコンテンツ
  - 書籍版 ⇨ 再構成
  - 合格対策テキスト
    - 知識編:章立てを再構成:全27章
    - 暗記項目集:標準問題集の各章に対応
  - 模擬問題
    - 書籍版の模擬問題
    - 暗記項目集から新たに作成した模擬問題

# スマホ教育アプリの開発 / 販売

## ■ 搭載した機能

### ▪ 学習機能

「合格対策テキストを読む」 → 知識編/ 暗記項目集

「模擬問題集を解く」 → 【知識編】 回答に必要な知識を確認

→ 【暗記項目集】 繰り返して効率よく暗記

→ 【模擬問題（4 択問題）】

### ▪ 学習進捗チェック機能

→ 既読箇所と理解度を★の個数で表示

→ 模擬問題の分野ごとに正解率を表示

→ 模擬問題の各問題について、前回の解答結果を○×で表示

対話型AIをCAEの教育に使えないかを検証



スマホ教育アプリのコンテンツで学習させる試み

# 対話型AI

## < 近年の動向 >

- モデルサイズとデータサイズの増加
- Large Language Modelの開発と応用
- OpenAI, Google, Meta AIなどのメガプラットフォームが開発
- Chat GPTの出現が対話型AIの活用に拍車をかけた
- 学習済みのモデルをファインチューニングし、自社のユースケースへ応用

# LARGE LANGUAGE MODEL (LLM)

## 大規模なデータセットを用いて学習された大規模なモデル

数テラトークン

数十テラパラメータ

### < 特徴 >

- 複雑なタスクを解決する強力な能力を示す
- Transformerアーキテクチャ
- 数千台のGPUを用いた分散学習
- 以下の三点をバランスよく増加させることが性能向上に繋がる (Scaling law)
  - データサイズ
  - モデルサイズ
  - 計算量
- **従来の言語モデルにはない能力が発現**
  - In-context learning  
与えられた例から問題解決の方法を勝手に学ぶ
  - Instruction following  
小規模データでマルチタスク学習を行うと未知の問題設定にも対応可能
  - Step-by-step reasoning  
最終的な答えに至るまでの中間ステップを用いると複雑なタスクを解くことが可能

# LLM迄の変遷

## Large language model (ChatGPT<sup>[6]</sup>, GPT4<sup>[7]</sup>, LLaMA<sup>[8]</sup>)

- ChatGPT
  - 人間との対話に特化
  - 実際に人間が作成した対話を元にFine-tuning
  - 公開二か月で月間利用者数が1億人
  - だれでも手軽にアクセスできる時代の幕開け
- GPT4
  - 入出力のマルチモーダル化（画像、音声）
  - ChatGPTよりも高い問題解決能力
  - 悪意のある返答や幻想などの低減
- LLaMA (Large Language Model Meta AI)
  - 7~65Bパラメータと比較的軽量ながらChatGPTに匹敵する性能
  - オープンソースで公開されており商用利用が可能

# LLAMA 2<sup>[14]</sup>

- Meta AIにより公開されるLLM
- Acceptable Use Policyに従う限りで学術・商用利用可能
- 7B/13B/70Bと様々なサイズのモデルが公開
- 日本語データセットでFine-tuningしたELYZA-japanese-Llama-2-7b<sup>[15]</sup>も公開されている



**「ELYZA-japanese-Llama-2-7b」を家庭用のGPUで動かす**

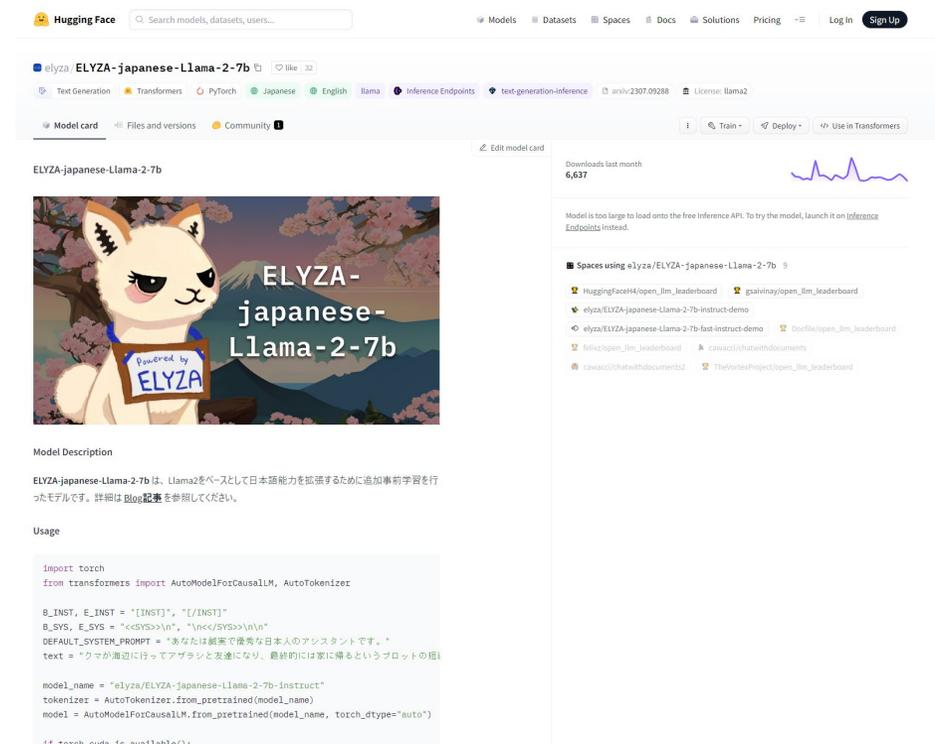
# ELYZA-JAPANESE-LLAMA-2-7B

## < 環境 >

- OS: Ubuntu 20.04 LTS
- RAM容量: 32GB
- GPU: NVIDIA GeForce RTX3090 24GB
- 仮想環境: Docker
- 仮想環境イメージ: [nvcr.io/nvidia/pytorch:23.04-py3](https://nvcr.io/nvidia/pytorch:23.04-py3)
- Pythonバージョン: 3.8.10

## < Pythonライブラリ >

- PyTorch 2.1.0a0+fe05266
- transformers 4.31.0



The screenshot shows the Hugging Face model page for `elyza/ELYZA-japanese-Llama-2-7b`. The page features a model card with a cute fox illustration and the text "ELYZA-japanese-Llama-2-7b" and "Powered by ELYZA". The model description states: "ELYZA-japanese-Llama-2-7b は、Llama2をベースとして日本語能力を拡張するために追加事前学習を行ったモデルです。詳細は [Blog記事](#) を参照してください。" The usage section includes a code snippet for inference:

```
import torch
from transformers import AutoModelForCausalLM, AutoTokenizer

B_INST, E_INST = "[INST]", "[/INST]"
B_SYS, E_SYS = "<SYS>[n]", "</SYS>[n]"
DEFAULT_SYSTEM_PROMPT = "あなたは最も優秀な日本人のアシスタントです。"
text = "クマが海辺に行ってアザラシと友達になり、最終的には家に帰るというプロットの短"

model_name = "elyza/ELYZA-japanese-Llama-2-7b-instruct"
tokenizer = AutoTokenizer.from_pretrained(model_name)
model = AutoModelForCausalLM.from_pretrained(model_name, torch_dtype="auto")

if torch.cuda.is_available():
```

# ELYZA-JAPANESE-LLAMA-2-7B

## < 手順 >

- 右図のプログラムを作成し実行
- 学習済みモデルは自動的にダウンロードされる

```
import torch
from transformers import AutoModelForCausalLM, AutoTokenizer

B_INST, E_INST = "[INST]", "[/INST]"
B_SYS, E_SYS = "<<SYS>>\n", "\n</SYS>>\n\n"
DEFAULT_SYSTEM_PROMPT = "あなたは誠実で優秀な日本人のアシスタントです。"

def main():
    model_name = "elyza/ELYZA-japanese-llama-2-7b-instruct"
    tokenizer = AutoTokenizer.from_pretrained(model_name)
    model = AutoModelForCausalLM.from_pretrained(
        model_name,
        torch_dtype="auto"
    )

    if torch.cuda.is_available():
        model = model.to("cuda")
        pass

    while True:
        text = input("##### プロンプトを入力してください: ")
        if text.strip() == "q":
            break

        prompt = "{bos_token}{b_inst} {system}{prompt} {e_inst} ".format(
            bos_token=tokenizer.bos_token,
            b_inst=B_INST,
            system=f"{B_SYS}{DEFAULT_SYSTEM_PROMPT}{E_SYS}",
            prompt=text,
            e_inst=E_INST,
        )

        with torch.no_grad():
            token_ids = tokenizer.encode(
                prompt,
                add_special_tokens=False,
                return_tensors="pt"
            )

            output_ids = model.generate(
                token_ids.to(model.device),
                max_new_tokens=512,
                pad_token_id=tokenizer.pad_token_id,
                eos_token_id=tokenizer.eos_token_id,
            )
            pass

        output = tokenizer.decode(
            output_ids.tolist()[0][token_ids.size(1):],
            skip_special_tokens=True
        )

        print(f"##### 出力: {output}")
        pass

if __name__ == "__main__":
    main()
    pass
```

# ELYZA-JAPANESE-LLAMA-2-7B

リソース

- GPUメモリは15GB程度確保できれば実行可能
- 推論時間は10秒程度



家庭用のGPUがあればだれでもローカル環境でLLMを動かすことができる！



# 弊社事例 I: ELYZA-JAPANESE-LLAMA-2-7B

## < Fine-tuning >

- 弊社のHPから簡易的なデータセットを作成
- QLoRA<sup>[16]</sup>と呼ばれる手法を用いてFine-tuning
  - メモリ消費量を抑えて学習
  - 10GBのGPUがあれば実行可能
  - 小規模のデータセットであれば10分程度で損失が収束

```
{
  "input": "Meshmanとは何ですか？",
  "output": "Meshmanは、CAEシステムを使用した経験のない設計者を対象とした、設計者が自立して使えるようになることを目指すCAEシステムです。"
},
{
  "input": "どうしてMeshmanを使うとCAEに習熟できるのですか？",
  "output": "Meshmanは手順を示すという形で、ユーザーの操作のお手伝いをしますが代行して自動実行はしません。ユーザーが何(What)をしたいかという要望を元に、"
},
{
  "input": "Meshman_FiberPackingとは何ですか？",
  "output": "短繊維モデル生成ソフトです。炭素繊維強化プラスチック(CFRP)のような繊維強化プラスチック複合材料の、確率均質化法による損傷進展解析を行う際"
},
{
  "input": "Meshman_ParticlePackingとは何ですか？",
  "output": "粉体がランダムに配置された空間に流体が流れるのを解析したい！ - 粉体の配置を色々変更して影響を確認したい！ - 個別要素法 (DEM) などの"
},
{
  "input": "Meshman_FEMとは何ですか？",
  "output": "構造・熱解析統合ソフトです。ADVENTUREプロジェクトよりリリースされたソフトウェアをベースに、初心者にも優しい手順ガイドや強力な結果可視化"
},
{
  "input": "Meshman_ParticleViewer_HPCとは何ですか？",
  "output": "Meshman_ParticleViewer_HPCは粒子法、差分法、ボクセル法向けにSTLファイルから生成された格子モデルを表示するソフトウェアです - 粒子法、"
},
{
  "input": "Meshman_Nastran_I/F for ADVENTUREとは何ですか？",
  "output": "Meshman_Nastran_I/F for ADVENTUREはNastranのバルクデータをADVENTUREのネイティブ(バイナリ)データに変換するソフトです。既存のNastran"
},
{
  "input": "株式会社インサイトについて教えてください。",
  "output": "インサイトは平成13年5月12日付けで有限会社から株式会社組織変更しました。インサイトは老若男女を問わず誰でも非常に簡単にコンピュータを"
},
{
  "input": "株式会社インサイトの会社設立の経緯を教えてください。",
  "output": "インサイトは私が平成8年10月から平成11年9月まで東京大学大学院工学系研究科システム量子工学専攻博士課程で矢川元基教授のもとで研究した成果を"
},
{
  "input": "株式会社インサイトが適用する技術は何ですか？",
  "output": "インサイトの扱うソフトウェアは複数の分野の技術を用いて開発されます。最も中核となる技術はソフトウェアエージェントと呼ばれる一種の人工知能"
},
{
  "input": "株式会社インサイトの商品について教えてください。",
  "output": "当社は大学のプロジェクトへの参加や他社との提携によりソフトウェアを供給しながら徐々に単独のパッケージ商品やウェブ上のサービスを展開して行"
}
```

# 弊社事例I: ELYZA-JAPANESE-LLAMA-2-7B

## < 実行例 >

##### プロンプトを入力してください: 株式会社インサイトとは何ですか？

##### 出力: 私はインサイト株式会社の一員として、人工知能や機械学習の技術を使用したソフトウェアの開発に携わっております。特に、エンジニアリングソフトの自動化、およびデータを使用した意思決定のサポートをするソフトウェアの開発に力をいれております。

##### プロンプトを入力してください: Meshman\_Nastran\_I/F for ADVENTUREとMeshman\_ParticleViewer\_HPCの用途の違いを比較し、簡単にまとめてください。

##### 出力: Meshman\_Nastran\_I/F for ADVENTUREとMeshman\_ParticleViewer\_HPCはそれぞれ異なるing tasksを行います。

Meshman\_Nastran\_I/F for ADVENTUREは、CAEシステムとの仲立ちをするソフトウェアです。手順としては、CAEシステムの予め定義されたフォーマットのデータを改変し、Meshman\_Nastran\_I/F for ADVENTUREの出力フォーマットに変換します。

Meshman\_ParticleViewer\_HPCは、粒子法、差分法、ボクセル法向けにSTLファイルから自動生成された格子を表示するソフトウェアです。手順としては、粒子法、差分法、ボクセル法向けにSTLファイルから格子を作成し、Meshman\_ParticleViewer\_HPCの出力フォーマットに変換します。

# QLORA : QUANTIZED LOW RANK ADAPTERS

必要なメモリ消費量を小さく抑えつつLLMのファインチューニングを行う手法

ネットワークのパラメータの量子化

- 4 bit Normal Float量子化
- 二重量子化



Low Rank Adapters

事前学習済みパラメータからの差分を計算することでFine-tuningを行う

$$h = W_0x + \mathbf{B}\mathbf{A}x, \quad \mathbf{B} \in \mathbb{R}^{d \times r}, \mathbf{A} \in \mathbb{R}^{r \times k}$$



市販の家庭用GPUでLLMのファインチューニングも可能！

# FINE-TUNING (発展)

## 固体力学2級模擬問題集をもとに問題と答えのペアを作成しFine-tuningを実行

```
{
  "input": "関数 $f(x,y)=\sin\{x^2+y\}$ を、点 $(0,0)$ の回りで一次迄テイラー展開したとき、正しいものは次のうちどれか? 1.  $f(x,y)=1$  2.  $f(x,y)=x$  3.  $f(x,y)=x+2y^2$  4.  $f(x,y)=y$ ",
  "output": "正解は4の、「 $f(x,y)=y$ 」です。"
},
{
  "input": " $A, B$ は $n$ 次正方行列、 $c$ は実数とする。行列式 $(|A|=\det\{A\})$ の性質で正しい物は次の内どれか。 1.  $|cA|=|c|^n|A|$  2.  $|cA|=c^n|A|$  3.  $|AB|=|A|^n|B|$  4.  $|AB|=|A^T|^n|B|$ ",
  "output": "正解は2の、「 $|cA|=c^n|A|$ 」です。"
},
{
  "input": "関数 $f(x)$ の1階微分 $f'(x)$ の中央差分と後退差分の式がこの順序でそれぞれ正しく記述されている組み合わせは①～④のどれか。 ①式A.  $f'(x)\approx\frac{-f(x-h)+f(x)}{h}$  式B.  $f'(x)\approx\frac{f(x+h)-f(x-h)}{2h}$  式C.  $f'(x)\approx\frac{f(x+2h)-f(x-2h)}{h}$  式D.  $f'(x)\approx\frac{f(x-h)+f(x)}{h}$  式E.  $f'(x)\approx\frac{f(x+2h)-f(x)}{h}$  1. C, A 2. E, D 3. B, A 4. B, E",
  "output": "正解は3の、「B、A」です。"
},
...
```

# FINE-TUNING

## < 正解率 >

20問 / 63問 ⇒ 31.7%

ランダムレート (25%)を多少上回る正解率

## < 所感 >

LLMのファインチューニングは参入障壁は低いが、実務に耐えうる性能を引き出すためには更なる改善が必要で、ここから先は相当労力のかかる作業となるのではないか

# 追加例 (正解したもの)



熱伝導率 $\lambda$ に関して、次のような主張が有る。ア:物体中の任意の点に於いて、微小な面要素を考える。熱伝導により、この面要素を単位時間、単位面積当たりに流れる熱量を $q$ 、面要素と垂直方向の温度の変化率を $\frac{d\theta}{dn}$ とする時、 $\lambda$ は、 $q$ と $-\frac{d\theta}{dn}$ に関する比例関係の比例定数である。イ:金属の一般的性質として、電気の伝わり易さと熱の伝わり易さの間には、ローレンツの法則と呼ばれる比例関係がある。ウ:一般に保温材の熱伝導率は、材料及び空気の熱伝導率の温度依存性や、場合に依っては、対流や放射の影響に因り温度と共に上昇する。ア、イ、ウの主張を正しく評価している物は次の内どれか。理由とともに答えてください。

- 1. ウのみが正しく、ア、イは誤りである。
- 2. ア、ウが正しく、イが誤りである。
- 3. ①~③のいずれでも無い。
- 4. ア、イ、ウの全てが正しい。



正解は4の、「ア、イ、ウの全てが正しい。」です。

根拠となる文章は1の、「熱伝導率 $\lambda$ に関して、次のような主張が有る。」です。

# 追加例 (正解したもの)



汎用構造解析プログラムを用いて弾性応力解析を行う際、入力データに於いて長さの単位としてmm、ヤング率の単位としてMPaを設定した場合、計算結果として得られる応力の単位は、次の内何れか。

- 1. N
- 2.  $\frac{N}{\text{mm}^2}$
- 3.  $\frac{\text{kgf}}{\text{mm}^2}$
- 4. GPa



正解は2の、「 $\frac{N}{\text{mm}^2}$ 」です。

# 追加例 (正解したもの)



$A$ 、 $B$ は $n$ 次正方行列、 $c$ は実数とする。行列式( $|A| = \det A$ )の性質で正しい物は次の内どれか。

- 1.  $|cA| = |c|^n |A|$
- 2.  $|cA| = c^n |A|$
- 3.  $|AB| = |A|^n |B|$
- 4.  $|AB| = |A^T|^n |B|$



正解は2の、「 $|cA| = c^n |A|$ 」です。

# 追加例 (正解したもの)



1階微分を前進差分で近似したものとして正しい式はどれか？

- 1.  $f'(x) \approx \frac{f(x) - f(x-h)}{h}$
- 2.  $f'(x) \approx \frac{f(x+h) - f(x-h)}{2h}$
- 3.  $f'(x) \approx \frac{f(x+2h) - f(x)}{h}$
- 4.  $f'(x) \approx \frac{f(x+h) - f(x)}{h}$



正解は4の、「 $f'(x) \approx \frac{f(x+h) - f(x)}{h}$ 」です。

# 追加例 (正解したもの)



正方行列  $A$  について、 $A$  が正則であることと同値な命題はどれか？

- 1.  $|A| = 0$
- 2.  $|A| = 1$
- 3.  $|A| > 0$
- 4.  $|A| \neq 0$



$|A|$  正解は4の、「 $|A| = \nexists x |Ax| = 0$ 」です。

# 追加例 (正解したもの)

 関数  $f(x, y) = \sin(x^2 + y)$  を、点  $(0, 0)$  の回りで一次迄テイラー展開したとき、正しいものは次のうちどれか?

- 1.  $f(x, y) = 1$
- 2.  $f(x, y) = x$
- 3.  $f(x, y) = x + 2y^2$
- 4.  $f(x, y) = y$

 正解は4の、「 $f(x, y) = y$ 」です。

# 追加例 (正解したもの)



一辺の長さが15mmの正方形断面で、長さ800mmの片持ち梁に、以下の3通りの荷重のどれかを負荷する。下の選択肢の内、正しい物を選べ。荷重ケース1:自由端に10Nの集中荷重。荷重ケース2:固定端部から長さの2/3の位置に25Nの集中荷重。荷重ケース3:固定端部から長さの1/3の位置に30Nの集中荷重。

- 1. 最大応力の絶対値が最も大きいのは、荷重ケース1である。
- 2. 最大応力の絶対値が最も大きいのは、荷重ケース3である。
- 3. 最大応力の絶対値が最も大きいのは、荷重ケース2である。
- 4. 最大応力の絶対値は、各荷重ケースで差が無い。



正解は3の、「最大応力の絶対値が最も大きいのは、荷重ケース3である。」です。

# 追加例 (不正解だったもの)



関数  $f(x)$  の1階微分  $f'(x)$  の中央差分と後退差分の式がこの順序でそれぞれ正しく記述されている組み合わせは1~4のどれか。

式A.  $f'(x) \approx \frac{-f(x-h)+f(x)}{h}$  式B.  $f'(x) \approx \frac{f(x+h)-f(x-h)}{2h}$  式C.  $f'(x) \approx \frac{f(x+2h)-f(x-2h)}{h}$   
式D.  $f'(x) \approx \frac{f(x-h)+f(x)}{h}$  式E.  $f'(x) \approx \frac{f(x+2h)-f(x)}{h}$

- 1. C、A
- 2. E、D
- 3. B、A
- 4. B、E

COPYRIGHT (C) 2023 | NSRF | JIN |  正解は4の、「 $f'(x) \approx \frac{f(x-h)+f(x)}{h}$ 」です。

# 追加例 (不正解だったもの)



微分演算子 $\nabla$ (ナブラ)の定義はどれか？

- 1.  $\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)$
- 2.  $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
- 3.  $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$
- 4.  $\left(\frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial z^2}\right)$



正解は2の、「 $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ 」です。

# 追加例 (不正解だったもの)



微分演算子gradの定義はどれか？

- 1.  $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
- 2.  $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$
- 3.  $\left( \frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z} \right)$
- 4.  $\left( \frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial z^2} \right)$



正解は2の、「 $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$ 」です。

# 追加例 (不正解だったもの)



連立一次方程式の解法は、直接法と反復法に分けることができる。次の(A)から(E)迄の5つの連立一次方程式解法のうち、反復法に分類されるものの組み合わせを選べ。(A)ウェーブフロント法 (B)CG法 (C)ガウスザイデル法 (D)GMRES法 (E)スカイライン法

- 1. DE
- 2. ACE
- 3. ABD
- 4. BCD



正解は3の、「ABD」です。

反復法に分類される方程式の例としては、三角分解法やガウスザイデル法が挙げられます。

# 追加例 (不正解だったもの)



担当の計算力学技術者のミスで解析上のトラブルが生じ、仕事に遅れが生じた。解析結果が出たが、会社所有のポストプロセッサでは、顧客の要求仕様に合致する後処理が出来ない事が判明した。手作業でやれば、仕様は満足させられるが、手間が掛かり過ぎて納期には間に合わない。この契約は納期を超過すると賠償請求される契約となっており、何としてもそれは避けたい。偶々担当の計算力学技術者が趣味で高機能後処理ソフトを購入し、私有のノートPCにインストールしていたのでそれを利用すると納期に間に合う事が分った。しかし、会社の内規で個人用PCに業務用データをコピーする事は禁じられている。これに対する次の対策のうち倫理的に最も妥当な物はどれか。

- 1. 上司の特別の許可を得て、自分の個人用PCに業務用データをコピーして後処理を納期迄に済ませる。
- 2. 特別に賠償責任無しで納期を延長出来ないかどうか顧客と調整する。
- 3. 人海戦術で手作業でも納期に間に合うように作業する事を上司に掛け合う。但し賠償金額相当の外注費用が発生する。
- 4. 上司に相談すると大事になるので、無断で自分の個人用PCに業務用データをコピーして後処理を納期迄に済ませる。図を見ると会社所有のソフトで無い事が明らかだか気づかなかった事にして貰う。



正解は4の、「上司に相談すると大事になるので、無断で自分の個人用PCに業務用データをコピーして後処理を納期迄に済ませる。図を見ると会社所有のソフトで無い事が明らかだか気づかなかった事にして貰う。」です。

# 追加例 (不正解だったもの)



コンピュータで次の整数同士の演算を行ったら、得られる結果として正しい物は何れか。 $2 \times 3 - (4 + 2 \div 3) \times 2 - 1$

- 1. 7
- 2. -4.2
- 3. -4
- 4. -3



正解は1の、「7」です。

# 追加例 (不正解だったもの)



脆性材料及び延性材料で出来た連続体を有限要素法により応力解析した際、結果を纏める為に注目する応力として最も適切な物は次の内どれか?但し、連続体は、多軸応力状態に有り、(a)脆性材料、(b)延性材料とする。

- 1. (a)最大主応力 (b)最大剪断応力
- 2. (a)ミーゼス応力 (b)最大主応力
- 3. (a)最大剪断応力 (b)最大圧縮応力
- 4. (a)中間主応力 (b)共役剪断応力



正解は2の、「(a)ミーゼス応力 (b)最大主応力」です。

# 追加例 (不正解だったもの)



固体棒中の圧縮もしくは引張応力波の伝播速度は、次のうちのどの式で与えられるか。但し、縦弾性係数を $E$ 、質量密度を $\rho$ とする。

- 1.  $E/\rho$
- 2.  $E^2/\rho$
- 3.  $\sqrt{E/\rho}$
- 4.  $\sqrt{\rho^2/E}$



正解は2の、「 $E^2/\rho$ 」です。

# 追加例 (不正解だったもの):略称



ファイル転送により、遠隔地にあるコンピュータとデータの交換を行う方法として、正しいものは次のうちどれか。

- 1. SCP
- 2. RAID
- 3. NFS
- 4. Samba



正解は3の、「NFS」です。

# 追加例 (不正解だったもの)



離散化手法について述べた以下の選択肢の各文章の内、適切では無い物を選択せよ。更にその理由も述べよ。

- 1. 重み付き残差法に於いて、重み関数と座標関数を等しく選ぶのがガラーキン法である。
- 2. 差分法は、領域を格子状網目に分解し、格子点に於ける値を用いて、微分を差分に近似し、微分方程式を差分方程式として解く物である。
- 3. レイリー・リッツ法は、連続体の基礎方程式の近似解を $\bar{\theta}(x, y) = \sum_{i=1}^M a_i \phi_i(x, y) + \phi_0(x, y)$ と仮定した時、変分原理によって $a_i$ を決定し、近似解を求める方法である。
- 4. ガラーキン法は、残差(近似解と正解の差を表す物)に重み関数を掛けて、それを最小とするように近似解を求める方法である。



正解は3の、「レイリー・リッツ法」です。

この法は、近似解を $\bar{\theta}(x, y) = \sum_{i=1}^M a_i \phi_i(x, y) + \phi_0(x, y)$ と代換する事を利用して、近似解を求める方法である。

# 追加例 (不正解だったもの)



以下の記述のうち正しい物はどれか。

- 1. 熱伝導率の単位は、 $[W/m]$ である。
- 2. 定常熱伝導方程式は、物質の密度の影響を受ける項を持つ。
- 3. 熱伝導解析において、断熱境界条件は、明示的に与える必要が有る。
- 4. 熱伝導問題を扱う場合、通常大気と接する外部境界は、常温の空気との自然対流による熱伝達境界として扱う場合が多い。



正解は3の、「熱伝導解析において、断熱境界条件は、明示的に与える必要が有る。」です。

# 追加例 (不正解だったもの)



企業の製品設計における構造解析の必要性や目的として適切でない物は次の内どれか。

- 1. 概念設計では設計の詳細は決まっていない為、実験を代替出来る程度のシミュレーション精度は必要で無い場合が多い。
- 2. 設計各段階でシミュレーションを実施するのは、計算機及び人的資源の問題で一般的に行うべきでは無いと言われている。
- 3. 詳細設計の評価において、常に実験を代替出来る程度のシミュレーション精度が要求される訳では無い。
- 4. 破損原因の評価のシミュレーションは、企業において最も初期にシミュレーションが使われた目的であった。



正解は4の、「破損原因の評価のシミュレーションは、企業において最も初期にシミュレーションが使われた目的であった。」です。

# 追加例 (不正解だったもの)



微分演算子 $\Delta$ (ラプラシアン)の定義はどれか？

- 1.  $\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right)$
- 2.  $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$
- 3.  $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$
- 4.  $\left(\frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial y^2}, \frac{\partial^2}{\partial z^2}\right)$



正解は2の、「 $\frac{\partial}{\partial x} + \frac{\partial}{\partial y} + \frac{\partial}{\partial z}$ 」です。

# 引用文献

- [1] A. Vaswani et al., “Attention Is All You Need,” NeurIPS, 2017.
- [2] J. Devlin et al., “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” NAACL, 2019.
- [3] M. Lewis et al., “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension,” ACL, 2020.
- [4] A. Radford et al., “Improving Language Understanding by Generative Pre-Training,” 2018.
- [5] C. Raffel et al., “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” J. Mach. Learn. Res., 2020.
- [6] <https://openai.com/chatgpt>.
- [7] <https://openai.com/research/gpt-4>.
- [8] <https://ai.meta.com/llama/>.
- [9] D. Ziegler et al., “Fine-Tuning Language Models from Human Preferences,” arXiv preprint arXiv:1909.08593, 2019.
- [10] L. Ouyang et al., “Training language models to follow instructions with human feedback,” NeurIPS, 2022.
- [11] J. Kaplan et al., “Scaling Laws for Neural Language Models,” arXiv preprint arXiv:2001.08361, 2020.
- [12] O. Nov et al., “Putting ChatGPT’s Medical Advice to the (Turing) Test,” arXiv preprint arXiv:2301.10035, 2023.
- [13] JB Alayrac et al., “Flamingo: a Visual Language Model for Few-Shot Learning,” NeurIPS, 2022.
- [14] Y. Hong et al., “3D-LLM: Injecting the 3D World into Large Language Models,” arXiv preprint arXiv:2307.12981, 2023.
- [15] <https://huggingface.co/elyza/ELYZA-japanese-Llama-2-7b-instruct>.
- [16] T. Dettmers et al., “QLoRA: Efficient Finetuning of Quantized LLMs,” arXiv preprint arXiv:2305.14314, 2023.