



木幅アルゴリズムの学習システムの構築

正答判定結果保持とグラフ画像生成の修正

B4 小林紹子

November 20, 2025

目次

1 概要

2 カテゴリ別の解答欄の作成

目次

1 概要

2 カテゴリ別の解答欄の作成

前回までの進捗状況

- 1 ○ × 問題だけでなく, 数値の入力問題でも正答判定を可能化.
- 2 それぞれの問題の正解・不正解を保存.
- 3 React Flow によるグラフ描画の調整.

今回の進捗状況

- 1 問題のカテゴリ別に解答欄を作成.
- 2 問題文と画像生成の連携.
- 3 問題ページの分割.
- 4 サーバー選びの調査.

目次

1 概要

2 カテゴリ別の解答欄の作成

データベースにカテゴリ属性追加

修正前:

```
1 model Problem{
2   id Int @id @default(
      autoincrement())
3   text String
4   image String?
5   answer Int
6 }
```

変更後:

```
1 enum ProblemCategory {
2   TrueOrFalse           // ○×問題
3   Input                 // 入力問題
4   Choice                // 選択問題
5 }
6
7 model Problem{
8   id Int @id @default(autoincrement())
9   text String
10  category ProblemCategory @default(TrueOrFalse)
11  image String?
12  answer Int
13 }
```

⇒ カテゴリ別に解答欄を作成へ

case 文による解答欄の判別 I

```
1 function renderAnswerUI(p: Problem) {
2   switch (p.category) {
3     case ProblemCategory.TrueOrFalse: // ○×問題の場合
4       return (
5         <div>
6           <button
7             onClick={() => handleAnswer(p.id, 1)}
8             style={{ ... }}>○</button>
9           <button
10            onClick={() => handleAnswer(p.id, 0)}
11            style={{ ... }}>×</button>
12          </div>
13        );
14 // 入力問題の場合は入力された値を保存・送信する関数 handleAnswer を作成.
15     case ProblemCategory.Input: // 入力問題の場合
16       return (
17         <InputProblem
18           p={p}
19           handleAnswer={handleAnswer}
20         />);
```


case 文による解答欄の判別 II

```
21  
22     case ProblemCategory.Choice: // 選択問題の場合  
23         //まだ選択肢を DB にどう入れるか未検討につき未完成  
24         return null;  
25  
26     default:  
27         return null;  
28 }  
29 }
```