



木幅アルゴリズムの学習システムの構築

問題表示画面作成

B4 小林紹子

December 13, 2025



目次

1 概要

2 データベースの構成を変更

3 現在の問題表示画面作成状況

4 今後の予定



目次

1 概要

2 データベースの構成を変更

3 現在の問題表示画面作成状況

4 今後の予定

前回までの進捗状況と課題

- ① ◎ × 問題は正答判定可能だが、数値の入力問題でも正答判定が不可能。
- ② それぞれの問題の正解・不正解が保存されない。
- ③ React Flow で思ったようにグラフが描けない。
- ④ 問題文と画像生成が連携されていない。
- ⑤ 問題がページ分割されずに一覧で出てくる。
- ⑥ サーバーをどれにするか調査する。

今回の進捗状況

- ① ○ × 問題以外にも対応できる解答欄作成.
- ② 問題文と画像生成が連携.
- ③ 問題を一覧でなくページ分割させる.
→ 大幅なデータベース構成の変更とコンポーネント作成を行う.



目次

1 概要

2 データベースの構成を変更

3 現在の問題表示画面作成状況

4 今後の予定

データベース相関図 |

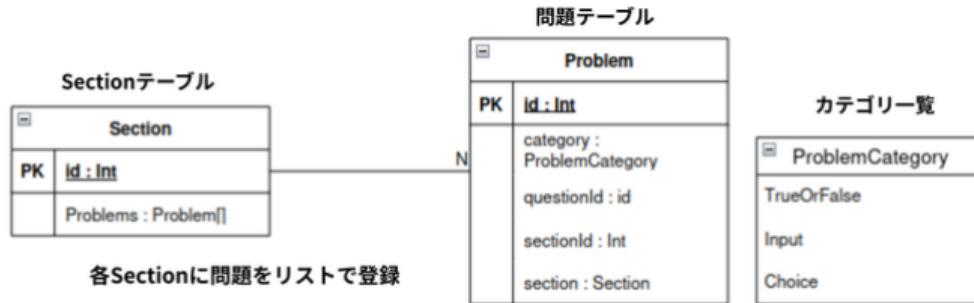


Figure: Section と Problem の関係

データベース相関図 II

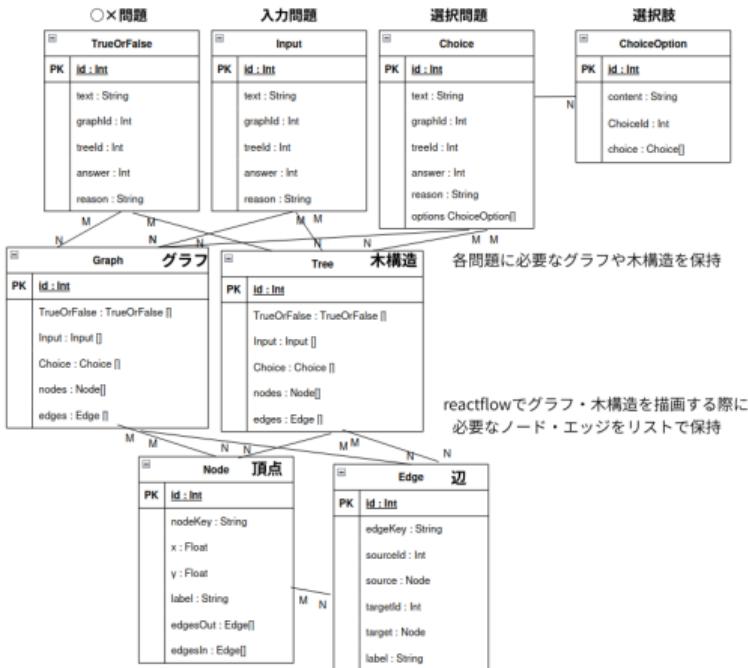


Figure: 個々の問題と画像の関係



問題取得までの流れ

- ① Section テーブルから Problem テーブルで問題のリストを取得.
- ② Problem テーブルの category と questionId でカテゴリ別のテーブルを参照.
⇒ category 別に API でテーブルを選択し,questionId をそのテーブルの主キーとして参照させる.
- ③ それぞれの問題から問題文, 答え, グラフなどのデータを表示.
- ④ グラフや木構造はノードやエッジの情報から ReactFlow で描画させる.

データベース変更により前回までの課題 All Clear

- ① Section ごとにページ分割 (Section テーブル作成) .
- ② 問題文と画像を連携 (各カテゴリ別の問題テーブルと Graph & Tree テーブルを外部キーで接続) .
- ③ 回答欄をカテゴリ別に適したものを表示 (カテゴリ別に問題テーブルを作成) .



目次

1 概要

2 データベースの構成を変更

3 現在の問題表示画面作成状況

4 今後の予定

問題の表示



Figure: 各問題の配置

グラフと木構造の表示

http://localhost:3000/problems/1

Section 1

```
graph LR; v1 --- v2; v2 --- v3;
```

```
graph TD; e1 --- e2;
```

問題 1

このグラフは外平面グラフである。

problems/\${number}が Section テーブルの id と一致.

○×問題の場合



問題 1

このグラフは外平面グラフである。

○



問題 1

このグラフは外平面グラフである。

○

正解！



入力問題の場合

問題 2

この木幅の値はいくつですか？



問題 2

この木幅の値はいくつですか？

不正解

選択問題の場合



問題 3

この木構造は全ての頂点を含んでいますか？

条件を満たしている 条件をみたしていない どちらともいえない

送信



問題 3

この木構造は全ての頂点を含んでいますか？

条件を満たしている 条件をみたしていない どちらともいえない

送信

不正解

ページ遷移



Prev

Next

それぞれ Section が存在する場合のみボタンが登場.



目次

1 概要

2 データベースの構成を変更

3 現在の問題表示画面作成状況

4 今後の予定



今後の予定

- 問題追加画面作成.
- データベースの構成変更.
- CSS に凝る.