

软工思考题参考答案

Problem 1

- `f1` 返回 `number` 类型
- `f2` 返回 `undefined` 类型 (`void` 也算正确)
- `f3` 返回 `number` 类型
- `f4` 返回 `Promise<number>` 类型 (`Promise` 或者 `object` 等也算正确)

Problem 2

会, 因为不能对 `interface` 使用 `instanceof`。

具体而言, `interface` 是 TypeScript 特有的关键字而不存在于 JavaScript 中, 其仅仅在编译时用于确定类型, 最后编译出来的 JavaScript 代码不会留下 `interface`。所以对 `interface` 使用 `instanceof` 无法正确编译。

Problem 3

这是因为 `&&` 具有**短路运算**的性质, 前项为假的时候后项不会计算。

Problem 4

其直接修改 `board` 本身, **未改变其内存地址**, 不触发 React/Next.js 的状态更新**重渲染**。

从修改只读变量导致运行时错误角度作答也算正确。

Problem 5

- `setBoardCache` 在 `board`, `id`, `dispatch` 发生变化或组件卸载, 且 `id` 为 `undefined` 时调用 (发起新一次的副作用)
- `resetBoardCache` 在 `board`, `id`, `dispatch` 发生变化或组件挂载, 且 `id` 为 `undefined` 时调用 (清除上一次的副作用)

依赖项不写 `dispatch` 亦可, 这是一个函数的引用, 理论上不应发生变化, 然而遵守 React Hook 规范需要在依赖项中写全所有可能的依赖。

以及请注意写全调用时机。

当用户在 Free Mode 时，离开该界面（切往其他界面或 Replay Mode）时会存储当前游戏状态到 Redux，再次进入 Free Mode 时，从 Redux 读取上次游戏状态并清理 Redux。

主要回答出用于保存 Free Mode 游戏状态即可，具体表述合理即算正确。

Problem 6

成功劫持，直接将 payload 解码即可得到用户名和密码的明文。

ChatGPT 也可能会犯错。请核查重要信息。

Problem 7

`source, destination` 表示重定向请求的接受模板和最终重定向到的目标。

该函数用于截获所有发往后端的 API 请求并将其重定向到真实的后端地址。

CICD 思考题

`--http` 选项可以直接处理 HTTP 请求而 `--socket` 则通过套接字与其他接受真实请求的服务器（如 Nginx）通信。

监听 `0.0.0.0` 可以接受来自包括外部的所有 HTTP 请求。