



300 0 0

300 : 300

ROUND 2 (0:00:17)

300 0 0

[DESCRIPTION](#) RULES README

CODEWRITING SCORE: 0/300

CodeFights supports different challenge types. One of them asks you to find a bug on a single line of the given code, usually referred to as a `DEBUGGING` challenge. Behind the scenes, we correctly implement each challenge ourselves and then use special comments with specific prefixes to introduce the buggy lines. Here is an image to help you visualize what that looks like:

```
1  bool pairOfShoes(std::vector<std::vector<int>> shoes) {
2      std::vector<int> leftShoes, rightShoes;
3      for (int i = 0; i < shoes.size(); i++) {
4          //DB 1//for (int i = 0; i + 1 < shoes.size(); i++) {
5          //DB 2//for (int i = 1; i < shoes.size(); i++) {
6              if (shoes[i][0] == 0) {
7                  leftShoes.push_back(shoes[i][1]);
8                  //DB 3//leftShoes.push_back(shoes[i][0]);
9                  //DB 4//leftShoes.push_back(shoes[i][2]);
10             } else {
11                 rightShoes.push_back(shoes[i][1]);
12                 //DB 5//rightShoes.push_back(shoes[i][0]);
13                 //DB 6//rightShoes.push_back(shoes[i][2]);
14             }
15     }
```

As you can see, each of the special comments looks like this:

```
<spaces>//DB <id>//<buggy line>
```

where `<spaces>` is a string consisting of zero or more spaces (for indentation), `DB` indicates that this comment is for a `DEBUGGING` challenge (let's assume this is the only type we support), `id` is a positive integer that helps us enumerate these and `<buggy line>` is some code that's almost identical to the line to be replaced but with an inserted bug (that is guaranteed not to contain any `'/'` symbols).

When importing these to our database, each of the special comments is used to create a debugging challenge. The importing script looks for the last non-special-comment line above the special comment and replaces it with `"<spaces><buggy line>"` while removing all the other special comments from the code. Examples below can help clarify this process further.

Your task is to produce `DEBUGGING` challenges given source code that includes the special comments and the `id` of the desired `DEBUGGING` challenge.

Example