

B : JFEN

原案 : tokoharu

テストケース : tokoharu

問題文 : kazu

解答 : tokoharu, kazu, not, mi_sawa

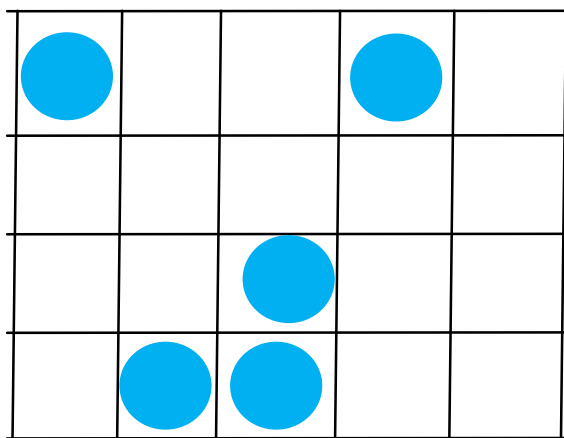
解説 : tokoharu

概要

- ▶ とあるゲームの盤面が与えられる。
 - ▶ 高々9*9の盤面にボールがあったりなかったり
- ▶ 一手動かすので、その盤面がどんなものか答えよ
- ▶ ただし、入出力は特定の文法に従う：ここが面倒
 - ▶ 次のページへ
- ▶ ところでこの問題の元ネタは SFEN(Shogi Forsyth-Edwards Notation) です。
 - ▶ 将棋AIは内部で盤面をこの形式で持っていることが多いです
 - ▶ 今回の問題はこれの状況をかなり簡単にしたものです

文法

- ▶ 次のようにして盤面と一対一に対応する
- ▶ 行ごとの情報を '/' 区切りで
- ▶ 空白マスはいくつ空白があるかの数で表現。 “33” のようなものはありません
- ▶ ボールは 'b' で表現。 “bb” などはありません。



b2b1
5
2b2
1bb2



b2b1/5/2b2/1bb2

解法

- ▶ 色々あるかもしれないがあまりラクはできなさそう。
- ▶ ...なので素直に
 - ▶ Jfenから盤面にデコード
 - ▶ 盤面の上で一手動かす
 - ▶ 盤面からJfenにエンコード
- ▶ とするのがよさそう
- ▶ 面倒そうだがそこまで大変なことにはならない

- ▶ 本問は解くために重要な考察ポイントはあまりない。
- ▶ 比較的簡単な変換規則の実装をどれだけ高速・正確に実装できるかが問われる

デコード例

```
//string jfenに入力盤面があり、これからboard[][]に書き込む。
int row = 0, col = 0;          // 何行目にいるか、何列目にいるか
for(int ind=0; ind<jfen.size(); ind++) {
    if(jfen[ind] == 'b') {     // ボールがあるとき
        board[row][col] ++ ; // 盤面のどこにボールがあるかメモ
        col++;                // 現地の情報がわかったのでcolを進める
    }
    else if(jfen[ind] == '/') {
        row++;                // 行の終端に来たので次の行へ
        col = 0;
    }
    else col += jfen[ind] - '0' ; // 空白マスは今読んでいる数だけ進む
}
```

エンコード例

```
string ans;
for(int i=0; i<row; i++) {
    int emp = 0;
    for(int j=0; j<col; j++) {
        if(board[i][j] == 1) {
            ans += "b";
            emp = 0;
        }
        else {
            // ボールがないとき
            if(emp == 0) ans += "0"; // 空白があるのでとりあえず数を入れる
            emp++;
            // emp更新
            ans[ans.size()-1]++; // ansの最後尾に今の空白マスの情報がある
        }
    }
}
if(i < row-1) ans += "/";
}
```

るので足す

もっときれいな書き方

- ▶ 以上の例は私のコードだが、洗練されていない印象を受けるためほかの人のコードも紹介したい。

```
vector<vector<char>> decode(const string &s){  
    vector<vector<char>> res(1);  
    for(auto &c : s){  
        if(c == '/')    res.emplace_back();  
        else if(c == 'b') res.back().emplace_back('b');  
        else            rep(_, c - '0') res.back().emplace_back(' ');  
    }  
    return res;  
}
```

```
string encode(const vector<vector<char>> &b){
    string res;
    for(auto &r : b){
        for(auto &c : r){
            if(c == 'b'){
                res.push_back('b');
            }else{
                if(!res.empty() and isdigit(res.back())) ++res.back();
                else res.push_back('1');
            }
        }
        res.push_back('/');
    }
    res.pop_back();
    return res;
}
```


統計情報

- ▶ FA : Sleep18000 (864sec)
- ▶ AC : 117 teams