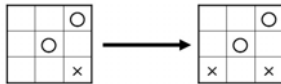


代数仕様言語 CafeOBJ による ボードゲームの形式的記述

情報システム工学科 3 年 063 吉川 大介

1. 概要

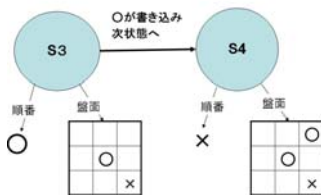
代数仕様言語 CafeOBJ を用いて、プレイヤーが互いに盤上にコマを置くタイプのボードゲームの振舞仕様を記述した。



2. ○×ゲームの概要

下図は○×ゲームの①交互の一つずつ、②盤上の、③空いているコマに書き込んでいく、というルールを図示したものである。

○×ゲームの振舞モデル



3. ○×ゲームの振舞仕様

以下に仕様の一部を示す。

```
ceq board(write(S, p1, xp, yp), xp', yp') =  
(if (xp == xp' and yp == yp') then 0  
else board(S, xp', yp') fi)  
if C-write(S, p1, xp, yp) .
```

項 `write(S, p1, xp, yp)` は、状態 `S` において `p1` が `(xp, yp)` 座標に `○` を書き込もうとした後の状態である。

上記の条件付き等式の左辺 `board(write(S, p1, xp, yp), xp', yp')` は、事後状態における `(xp', yp')` 座標のセルを表す。

条件部の項 `C-write(S, p1, xp, yp)` は、`write` 操作が可能かどうかを判定する。

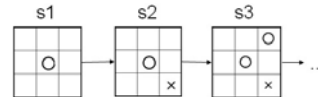
右辺では、さらに条件分岐しており、`(xp, yp)` 座標と `(xp', yp')` 座標が等しいときに `○` を返し、そうでないときに `board(S, xp', yp')`、すなわち `write` 操作以前の状態

`S` における `(xp', yp')` 座標を返す。

4. 仕様の実行

```
eq s1 = write(init, p1, 2, 2) .  
eq s2 = write(s1, p2, 3, 3) .  
red board(s2, 2, 2) .
```

初期状態 `init` の座標 `(2,2)` に `p1` が `○` を書き込んだのが状態 `s1`。状態 `s1` の座標 `(3,3)` に `p2` が `×` を書き込んだのが状態 `s2`。



`red` により項の簡約が行われ、指定された座標の盤面の状態が出力される。

```
-- opening module OX. done. _*  
-- reduce in %OX : (board(s2, 2, 2)):Cell  
(0):Cell  
(0.000sec_for_parse, 807rewrites (0.063  
sec), 1377 matches)
```

5. まとめ

- ×ゲームの仕様は定式化した。
- 定式化したときの `CELL` の数の変更やオペレータ追加の余地を残すことで、汎用性を高められた。

- 状態数が増えると、実行に時間がかかる。

例 : `red board(s9, 2, 2) .`

```
-- reduce in %OX : (board(s9, 2, 2)):Cell  
(0):Cell  
(0.000sec_for_parse, 409231rewrites (17.  
938 sec), 694769 matches)
```

6. 今後の課題

- 勝敗の決着や反則等の手順が途中で現れた場合、それを出力できるようにする。
- より高度なボードゲームの仕様を定式化していく。