

Java 2013 CW1

Материал из SEWiki

Контрольная работа (условие задачи)

Сегодня мы будем писать игру крестики-нолики на поле размером $N \times N$, где для выигрыша необходимо поставить К в ряд. Присутствующие классы:

1. Position - текущее состояние игрового поля. В игре крестики-нолики 1 - крестик, 2 - нолик, 0 - ничего нету. Метод toString должен выдавать состояние поля в виде таблицы чисел.
Например,
1100
0100
0222
0000
2. Player - ИИ игрока для какой-нибудь игры. Чтобы определить для какой именно есть метод getGameType возвращающий Class самой игры. getName - имя ИИ. move - делает 1 ход и возвращает НОВЫЙ Position (а не измененный старый)
3. XOPlayer - ИИ для игры XO.
4. Game - класс абстрактной игры принимающий в качестве параметров конструктора двух игроков. Если класс игры не совпадает и классом игры у Player, должен порождаться RuntimeException. Game реализует интерфейс Iterable, что позволяет итерироваться по позициям возникающим в процессе игры. В Game необходимо добавить метод позволяющий определить победителя.

Создайте класс Tournament<T extends Game> implements Iterable<T>, который будет производить турнир среди различных ИИ для конкретной игры. В качестве параметра конструктора Tournament принимает следующее:

1. Каталог с class-файлами ИИ
2. Class<T> gameClass
3. Object[] - набор параметров конструктора соответствующего Game.

При этом должны быть загружены только те ИИ, которые соответствуют данной игре. Должен быть предусмотрен метод, который выдает результат турнира - таблицу вида

```
Вася Петя Коля
Вася 1 1 0
Петя 2 0 0
Коля 0 1 1
```

Число - количество побед при двух партиях (в первой ходит первым первый игрок, во второй - второй)

во второй - второй)

Проведите турнир для игры ХО.

В данной работе не требуется написать эффективный ИИ. Задача правильно реализовать предложенную архитектуру (далеко не идеальную :)). Тем не менее, за хорошо написанный ИИ будут добавляться баллы.